Universidade Federal de Juiz de Fora
Programa de Pós-Graduação em Modelagem Computacional

Ferramenta Computacional para Apoio à Análise de Regressão de Dados

Por
Samuel Belini Defilippo

Juiz de Fora, MG – BRASIL
Agosto de 2008

Defilippo, Samuel Belini

Ferramenta Computacional para Apoio à Análise de Regressão de Dados / Samuel Belini Defilippo – 2008.

140 f. il.

Dissertação (Mestrado em Modelagem Computacional) - Universidade Federal de Juiz de Fora, Juiz de Fora, 2008.

1. Ciência da computação 2. Estatística. 3. Otimização (Matemática) 4. Otimização combinatória. I. Título

CDU 681.3

Ferramenta Computacional para Apoio à Análise de Regressão de Dados

Samuel Belini Defilippo

DISSERTAÇÃO SUBMETIDA AO PROGRAMA DE PÓS-GRADUAÇÃO EM MODELAGEM COMPUTACIONAL DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS (M.SC.) EM MODELAGEM COMPUTACIONAL.

Aprovada por:

Prof. Helio José Corrêa Barbosa, D.Sc.
(orientador)

Laurent Emmanuel Dardenne, D.Sc.

Carlos Cristiano Hasenclever Borges, D.Sc.

Juiz de Fora, MG – BRASIL
Agosto de 2008

"Não é o bastante ver que um jardim é bonito sem ter que acreditar que há fadas escondidas nele?"

Douglas Adams

AGRADECIMENTOS

Agradeço a todos que direta e indiretamente me ajudaram a concluir este trabalho. Especialmente ao meu Orientador e meu Co-orientador que foram muito importantes ao me ajudar e influenciar para o estudo aqui publicado.

Resumo da Dissertação apresentada à UFJF como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

FERRAMENTA COMPUTACIONAL PARA APOIO À ANÁLISE DE REGRESSÃO DE DADOS

Samuel Belini Defilippo
Agosto / 2008

Orientador: Helio José Corrêa Barbosa
Co-orientador: Henrique Steinherz Hippert

A análise de regressão de dados encontra aplicação em diversas áreas e o modelo obtido pode ser em seguida usado intensamente dentro de outro processo de otimização. Escolher o modelo que melhor se ajuste a um determinado banco de dados, contudo, ainda é um processo demorado, e muitas vezes heurístico. Neste trabalho foi desenvolvida uma ferramenta computacional de apoio a este processo de análise (escolha de modelos e a estimação dos parâmetros), baseado em um Algoritmo Genético aplicado aos modelos *Least Mean Squares*, *Multi Layer Perceptron* e *k-Nearest Neighbors*. A ferramenta é testada em diferentes bancos de dados, sendo um deles oriundo da área de desenho racional de fármacos baseado em estrutura, onde estão previstas aplicações futuras.

Abstract of Dissertation presented to UFJF as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

A COMPUTATIONAL TOOL TO SUPPORT NONLINEAR REGRESSION ANALYSIS

Samuel Belini Defilippo
August / 2008

Advisor: Helio José Corrêa Barbosa
Co-advisor: Henrique Steinherz Hippert

Regression analysis has application in several areas, and the models obtained can be used afterwards in optimization processes. Choosing the best model for a given databases, however, is still a time-consuming task, which is frequently done in a heuristic way. In this work we develop a computational tool to support the choice of models and the estimation of parameters, based on the application of a Genetic Algorithm to the three groups of models: Least Mean Squares ones, Multi Layer Perceptrons and k-Nearest Neighbors ones. The tool is tested in different databases, one of them originating from the area of structure-based rational drug design, where future applications are foreseen.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Siglas

AG	Algoritmo Genético
KNN	Algoritmo dos vizinhos mais próximos
LMS	*Least Mean Square*
MLP	Perceptron de Múltiplas Camadas
PCA	Análise de Componentes Principais
RBF	Função de Base Radial
RNA	Rede Neural Artificial
SVM	Máquina de Vetor Suporte

1. Introdução

Para introduzir o trabalho, primeiro iremos explicar os conceitos básicos para o entendimento do processo de reconhecimento de padrões. Do ponto de vista humano, podemos definir o reconhecimento de padrão o momento onde o observador se vê avaliando as características de uma pessoa e tenta prever o que ela irá fazer ou como ela irá se mostrar no dia a dia.

O ser humano involuntariamente faz esse tipo de processamento e está sempre tentando corrigir suas previsões. Desse exercício o ser humano consegue prever diversas situações, como se vai chover ou se o transito vai engarrafar.

Diferente do ser humano e vendo agora o ponto de vista do computador, o mesmo tenta responder a essas perguntas, porém de maneiras diferentes, pois nossos computadores não aprendem com a mesma facilidade do ser humano.

Esse conjunto de problemas se divide em duas classes básicas de previsão. Os classificadores e o regressores. Pelo dicionário Michaelis temos as seguintes definições:

- **clas.si.fi.ca.ção**: sf (classificar+ção) 1 Ação ou efeito de classificar. 2 Distribuição por classes. 3 Apreciação do mérito de alguém. 4 Ato ou efeito de classificar(-se) em concurso ou competição. 5 Posição em uma escala gradual de resultados de um concurso ou competição: Sua classificação foi ótima: 2º lugar. 6 Hist nat Arranjo sistemático ou método de arranjo de plantas e animais em grupos ou categorias de acordo com suas afinidades ou caracteres comuns. 7 Dir Definição jurídica da infração da lei, antes da fixação da pena. C. periódica dos elementos, Fís e Quím: quadro em que os elementos são dispostos em ordem crescente de seus números atômicos, formando grupos com propriedades químicas análogas.
- **re.gres.são:** sf (lat regressione) 1 Regresso, volta. 2 Retrocesso, reversão. 3 Biol Regresso de um tecido, de um indivíduo, a um estado anterior, menos aperfeiçoado. 4 Ret Figura em que se repetem as palavras na ordem inversa com diferente sentido: Deve-se comer para viver e não viver para comer. 5 Psicol Adoção, por uma pessoa, de comportamentos menos desenvolvidos que os normais, em sua idade. 6 Gram Efeito assimilador ou dissimilante de um fonema sobre outro anterior. R. fonética, Gram: volta à forma etimológica.

Em nosso contexto, as definições podem ser definidas como: Os classificadores nos ajudam a classificar um determinado conjunto de dados. Ou seja, a partir de características de um conjunto de indivíduos, tentamos separá-los e classificá-los em grupos de indivíduos simulares. Os regressores tentam associar um valor, contínuo, ao conjunto de características, não tendo assim um grupo contável de repostas esperadas.

Um exemplo simples de cada um seria: para classificadores, determinar o sexo de uma pessoa considerando seu peso e altura; para regressores, tentar prever quanto pesa uma pessoa tendo dados como sua altura e sexo.

Em nosso trabalho, vamos utilizar alguns modelos adaptativos de regressão para avaliação de dados e tentar chegar à melhor previsão.

Os modelos adaptativos são modelos que se adaptam aos dados, ou seja, que ajustam seus parâmetros com o objetivo de representar o comportamento dos dados. E essa linha tem como base mais comum a utilização dos modelos de Redes Neurais Artificiais (RNA), que fazem uso do conceito de aprendizado inspirado no funcionamento do cérebro humano. Iremos também utilizar outros modelos de regressão, como a regressão linear e o algoritmo K- vizinhos mais próximos (*k-nearest neighbours* - KNN), que utiliza uma medida de distância entre os dados para identificar seus vizinhos e a partir daí prever com base nos mesmos.

Os diversos processos de regressão dependem da definição de parâmetros importantes para chegar a um bom modelo, e a escolha desses valores vai influenciar na construção de um modelo mais complexo ou mais simples como também num modelo mais preciso ou mais geral. Encontrar esses valores pode ser uma tarefa difícil e demorada, pois a grande maioria dos modelos não é linear e conseqüentemente seus parâmetros podem se relacionar de forma muito complexa.

Na construção desses modelos utilizamos um processo de otimização (Algoritmo Genético, AG) que faz uso da programação evolucionista para estimar os parâmetros do modelo.

Outro conceito interessante que utilizamos em nosso trabalho é o conceito de filtro e seleção dos atributos dos dados de entrada. Esse processo faz uma análise dos dados de entrada e tenta prever quais os atributos são mais importantes.

Para isso utilizamos duas metodologias. A primeira é a Análise de Componentes Principais (PCA) que altera a organização dos dados fazendo com que a informação dos dados de entrada fique acumulada em um número menor de atributos, possibilitando assim a utilização de um número menor de atributos. A segunda está integrada ao AG, onde além dos parâmetros dos modelos, ele tenta também selecionar quais atributos irão participar do processamento, utilizando assim da informação de saída para definir qual é a melhor seleção de variáveis.

1.1. Problema e Motivação

A variedade de métodos para criarmos um modelo de regressão de dados nos leva para há um longo caminho de pesquisa. Além dos diversos métodos, a maioria deles possuem uma variedade de parâmetros que alteram o funcionando do mesmo. A escolha do modelo e de seus parâmetros é um grande problema.

Além disso, a análise dos dados de entrada é importante para a geração de um modelo mais simples, facilitando assim o processo computacional para execução dos modelos.

Nossa motivação então é tentar diminuir esse trabalho, criando uma ferramenta que automatiza o processo de definição dos parâmetros de cada modelo, como também o processo de seleção dos atributos dos dados de entrada, criando um comparativo entre os melhores modelos encontrados.

A partir desse comparativo, o pesquisador poderá definir melhor quais os passos a serem tomados em seguida para a criação de seu modelo de regressão.

Para validarmos e mostrarmos os resultados da ferramenta, utilizamos quatro banco de dados, cada um com suas características e diferenças para assim explorar as diversas facetas da regressão de dados.

1.2. Métodos

Basicamente o trabalho faz uso dos métodos de AG, RNA, KNN e PCA. O AG é utilizado na otimização dos parâmetros dos modelos, o RNA e KNN na regressão dos dados e o PCA na tentativa melhorar a informação de entrada.

Do AG iremos utilizar uma estruturação já bem conhecida que é o uso de um cromossomo binário com operadores de mutação e *crossover* de um ponto. Seu processo de seleção é feito através de torneios.

Do RNA fazemos uso de duas estruturas, uma delas é um *Least Mean Square* (LMS) onde não temos camada oculta e não temos parte não-linear. É comparado a uma regressão linear simples, porém lança mão do procedimento de *backpropagation* para corrigir seus pesos. Temos também o Perceptron de Múltiplas Camadas (MLP), que tem uma estrutura não-linear em uma camada interna.

Do KNN utilizamos a idéia de vizinho mais próximo com uma aproximação dada pela média ponderada pela distância, onde a menor distância tem um peso maior.

O PCA será utilizado na tentativa de diminuir as variáveis de entrada utilizadas nos modelos. Como ele nem sempre ajuda, sua execução é opcional.

O formato utilizado para a comparação dos resultados obtidos foi a criação de um gráfico "*boxplots*" que é a representação dos valores obtidos em caixas, onde a linha que divide a caixa representa a mediana dos dados, as extremidades da caixa os valores do com distância de um desvio padrão tanto para direita quanto para esquerda, a linha

contínua (que continua após a caixa) os valores com distância de dois desvios padrão e os pontos fora da linha os valores que estão a mais de dois desvios padrão.

1.3. Estrutura do Trabalho

Inicialmente iremos apresentar os banco de dados utilizados no trabalho. Isso será feito no capítulo 2. Nele iremos mostrar uma descrição dos banco de dados e fazer uma análise inicial de suas informações, criando condições para um melhor entendimento dos resultados.

Para um melhor entendimento relacionado aos modelos utilizados no trabalho, iremos no capítulo 3 fazer uma revisão sobre cada um deles, introduzindo a teoria de cada tentando assim facilitar a leitura do capítulo 4 que irá descrever o funcionamento completo da ferramenta.

Finalizando, no capítulo 5, analisamos os resultados obtidos e junto com o capítulo 6 fazemos as considerações finais sobre o trabalho.

2. Descrição dos problemas

Inicialmente tínhamos um problema para estudos de novos fármacos, e utilizamos de seus dados para validar os métodos da ferramenta. Porém vimos que era necessário a utilização de outros dados para criamos uma melhor apresentação do potencial da ferramenta.

Todos os dados são de eventos reais para tornar assim a análise do modelo mais próxima do dia a dia. Neste capítulo vamos fazer uma análise estatística descritiva dos dados e explicar a origem de cada um deles.

A normalização utilizada nos "*boxplots*" comparativos é a normalização gaussiana, onde subtraímos a média e dividimos pelo desvio padrão.

2.1. Base de dados dos Complexos Proteína-Ligante

O banco de dados vem do estudo de fármacos de Rêgo [1].

Diferentes do método histórico de descoberta de novos fármacos que utiliza de testes de tentativa e erro de substâncias químicas em animais, hoje o desenho racional de fármacos é realizado com o conhecimento das respostas químicas específicas no corpo ou no organismo alvo, e com a utilização destas informações para o ajuste de um tratamento ideal.

Como exemplo de desenho racional de fármacos pode-se citar o uso da informação tridimensional de biomoléculas que é obtido com técnicas como difração de Raios-X em cristais e espectrometria de Ressonância Magnética Nuclear e é referida normalmente como Desenho Racional de Fármacos Baseado em Estrutura.

Cada etapa desse processo requer tempo e investimento, sendo muito importante identificar o mais cedo possível os agentes que são provavelmente menos promissores, permitindo uma concentração de esforços nos compostos que têm maior probabilidade de chegar ao mercado.

Com o intuito de melhor entender e simular este mecanismo, métodos computacionais tornaram-se componentes cruciais de muitos programas utilizados na produção de fármacos, buscando, com técnicas de filtragem em bancos de dados, por ligantes ou estruturas ideais, como também no refinamento e otimização dos compostos previamente identificados.

A abordagem computacional possibilita a realização de testes que agilizam o processo manual gerando uma economia considerável nos custos relativos à produção de novos fármacos.

Nele temos 50 complexos proteína-ligante com 6 variáveis importantes: número de pontes de hidrogênio, energia eletrostática, energia de Lennard-Jones, porcentagem da superfície acessível ao solvente do ligante em contato com a proteína, número de ligações torcionáveis e número de ligações torcionáveis congeladas no processo de interação intermolecular.

Todas as variáveis são numéricas, sendo algumas inteiras e outras reais.

Tabela 2-1 – Variáveis dos dados de complexos proteína-ligante

Variáveis	Máximo	Mínimo	Média	Desv. Padr.	Descrição
B	98.1120	23.8630	80.8646	17.6964	Percentagem da superfície total do ligante em contato com a proteína
C	1.0876	-66.9633	-14.2673	16.9390	Energias de Lennard-Jones
D	-1.4073	-56.2872	-26.7257	11.8906	Energias eletrostática
E	14	0	4.8800	3.3481	Número de possíveis pontes de hidrogênio existentes entre a proteína e o ligante
F	14	0	4.2600	2.5778	Número de ligações torcionáveis
G	14	0	3.2600	2.6712	Número de ligações torcionáveis congeladas
Saída	10.8000	1.4900	5.5234	2.2272	log Ki ou - log Kd

Fonte: Dados calculados

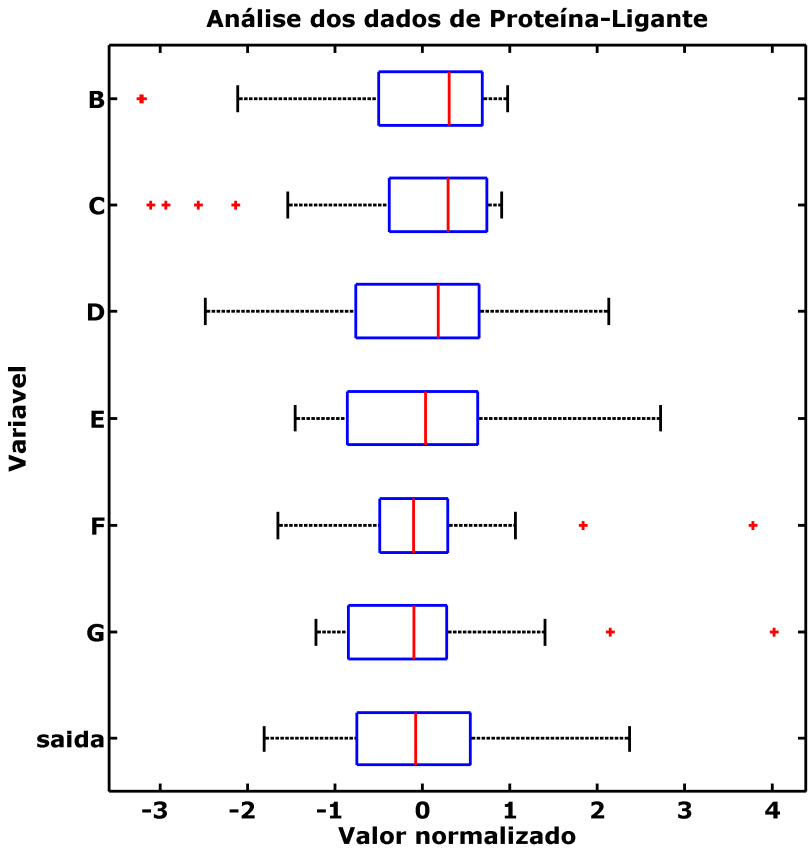

Figura 2-1 – Boxplot com os dados de Proteína-Ligante normalizados

2.2. Base de dados de Consumo de Energia

O banco de dados utilizado resultou de um levantamento organizado pelo Departamento de Estatística da UFJF em 2004-2005. Este banco contém as respostas a um questionário de 72 questões, obtidas em 557 residências que haviam sido escolhidas por meio de um processo de amostragem em várias etapas. As questões se relacionavam às características de cada residência (área construída, número de cômodos, etc.), às características sócio-econômicas de seus moradores (nível de escolaridade, número de carros, número de empregadas, etc.), aos hábitos de consumo dos moradores (freqüência e horário em que eram utilizados os diversos equipamentos elétricos existentes) e às atitudes frente a políticas de conservação de energia e de racionalização do consumo.

Após a conclusão do levantamento, uma equipe retornou a todas as residências onde haviam sido feitas as entrevistas e registrou suas coordenadas (latitude e

longitude) por meio de um aparelho de *Global Positioning System* (GPS). Estas coordenadas, em metros, foram incorporadas à planilha de dados.

As respostas aos questionários haviam sido digitadas originalmente numa planilha com 2070 variáveis (a maioria das quais binárias). Para reduzir este número, retiramos todas as variáveis relativas aos hábitos de consumo de energia e à conservação e racionalização do consumo; mantivemos apenas as variáveis que descreviam as características físicas da residência e o perfil sócio-econômico dos moradores. A Tabela 2-2 mostra as variáveis disponíveis e a Figura 2-2 um comparativo da dispersão dos dados.

Como haviam muitos valores faltantes entre as variáveis que mediam o consumo mensal, procuramos identificar os três meses consecutivos em que havia maior quantidade de dados disponíveis.

Adotamos então como variável indicadora do consumo a média entre estes três meses (março a maio de 2004). A seguir, eliminamos todos os casos com valores faltantes, com erros óbvios ou com registros duvidosos em qualquer variável; restaram 444 casos para análise.

Tabela 2-2 – Variáveis dos dados de consumo de energia

Variáveis	Máximo	Mínimo	Média	Desv. Padr.	Descrição
empregd	1	0			tem empregada doméstica?
analfa	1	0			é analfabeto
1inc	1	0			1º grau incompleto
1comp	1	0			1º grau completo
2inc	1	0			2º grau incompleto
2comp	1	0			2º grau completo
3inc	1	0			3º grau incompleto
3comp	1	0			3º grau completo
carros	5	0	0.6757	0.8025	número de carros
tipo1	1	0			se é casa
tipo2	1	0			se é apartamento
tipo3	1	0			outro tipo de residência
area1	1	0			área construída (0..50m^2)
area2	1	0			(51..75m^2)
area3	1	0			(76..100m^2)
area4	1	0			(101..150m^2)
area5	1	0			(151..200m^2)
area6	1	0			(acima de 200m^2)
comodos	20	2	6.8491	2.5216	número de cômodos
banheiro	7	1	1.7252	0.9316	número de banheiros
relogio	1	0			se tem relógio
mono	1	0			monofásico
bifase	1	0			bifásico
trifase	1	0			trifásico
resident	17	1	3.6329	1.7529	número de residentes
coord_n	7601199	7588406	7593900	2789.60	coordenadas norte (m)
coord_e	673269	661042	668690	2823	coordenadas este (m)
Saída	665.70	28	169.97	94.08	consumo em kWh

Fonte: Dados calculados

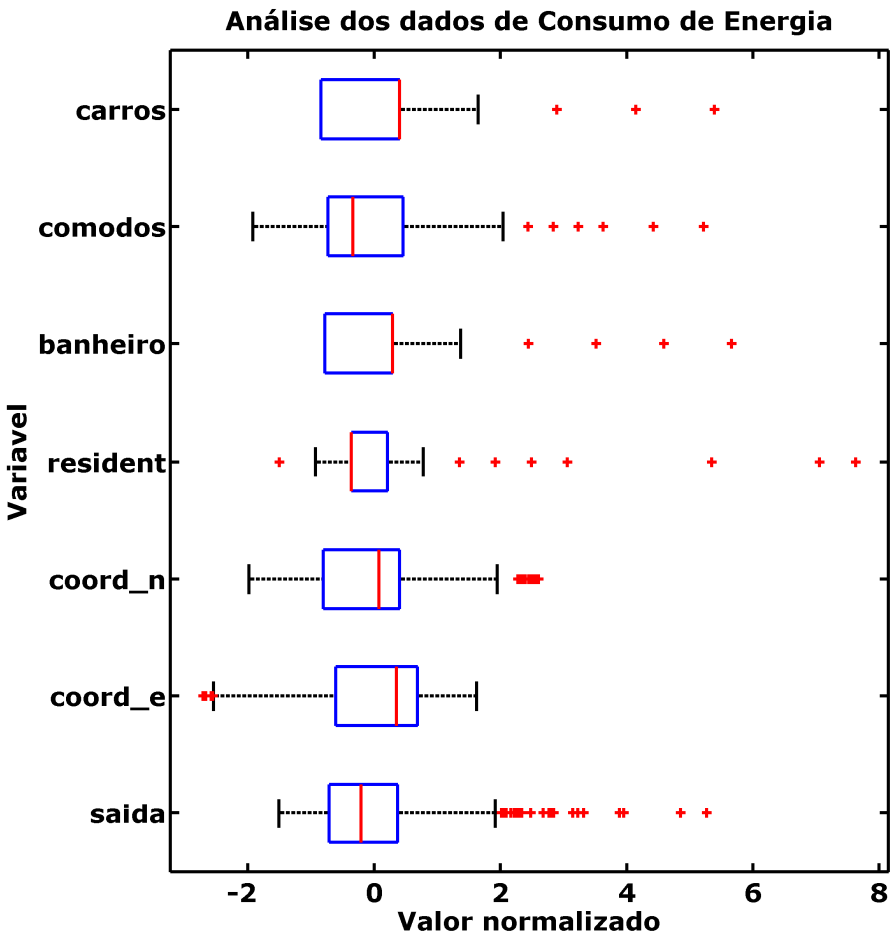

Figura 2-2 – Boxplot com os dados de consumo de energia normalizados

2.3. Base de dados de Preço de Imóveis

O banco de dados utilizado aqui é o mesmo da pesquisa realizada por Bráulio [2]. No planejamento realizado pela referida pesquisadora, antes da coleta dos dados foi contemplado o espaço físico, população a estudar e o número de imóveis a serem pesquisados.

Vale observar que, no mercado de imóveis, é freqüente a entrada de dados novos e, por isso, deve-se fazer um novo levantamento a cada nova avaliação para garantir a representação dos novos dados na amostra [3].

Os bancos de dados de apartamentos, casas e terrenos possuem 44, 51 e 24 dados de entrada respectivamente.

A variável de saída é o preço, que representa o valor de venda do imóvel em Reais. As variáveis originais estão relacionadas, classificadas e descritas em Tabela 2-3, Tabela 2-4 e Tabela 2-5 para os tipos de imóveis, apartamentos, casas e terrenos, respectivamente.

Tabela 2-3 – Variáveis dos dados de imóveis (apartamento)

Variáveis	Máximo	Mínimo	Média	Desv. Padr.	Descrição
pos. do apto	3	1	2.6364	0.6851	Identifica a posição do apartamento em relação ao prédio: 1 fundo, 2 lateral, 3 frente
elevador	2	0	1.2500	0.6862	Identifica a quantidade de elevadores no prédio.
garagem	4	1	1.4545	0.6631	Quantifica o número de vagas para carro disponível para cada apartamento.
local.	1	0			Identifica a existência ou não de lavanderia.
área	374	38	183.0464	84.3500	Corresponde à superfície ou área do apartamento expressa em metros quadrados, obtida do registro de imóveis.
pavimento	20	3	10.6364	4.5089	Indica o número de pavimentos do prédio.
andar	4	1	2.5000	0.7924	Identifica o andar que o apartamento está localizado. 1 primeiro, 2 segundo, 3 terceiro ou mais, 4 cobertura
peças	17	4	9.7273	2.8803	Quantifica as peças constituintes do imóvel.
salas	3	1	1.8636	0.7019	Indica o número de salas existentes no apartamento.
dormitório	3	1	2.2955	0.5532	Quantifica o número de dormitórios.
suíte	1	0	0.8409	0.3700	Identifica a presença ou não de suíte.
banheiro	3	1	1.2727	0.5440	Identifica o número de banheiro social.
dep. de emp.	1	0	0.6136	0.4925	Identifica a existência ou não de dependência de empregados.
dist. escola	3	1	2.7500	0.5757	Identifica a quanto o imóvel se localiza
dist. hosp.	3	1	2.3864	0.7840	próximo de escolas, supermercados,
dist. merc.	3	1	2.7045	0.6675	hospitais e do centro comercial. 1 mais de 800m, 2 de 500m a 800m, 3 até 500m
acab.	3	2	2.3636	0.4866	Identifica os vários níveis de acabamento. 1 baixo, 2 normal, 3 alto
revest. préd	4	1	3.0682	1.3494	Identifica o revestimento do prédio. 1 reboco/emboço, 2 tinta plástica, 3 pastilhas, 4 mármore
conservação	5	2	3.7273	0.7884	Identifica o nível de conservação do imóvel. 1 péssimo, 2 regular, 3 bom, 4 ótimo
idade real	5	1	3.2045	1.1119	Idade real: idade cronológica do edifício reflete o estágio tecnológico. 1 vinte anos ou mais, 2 de quinze a vinte anos, 3 de onze a quatorze anos, 4 de seis a dez anos, 5 de dois a cinco anos, 6 até um ano.
idade aparen	6	1	3.7955	1.4400	Idade real: idade cronológica do edifício reflete o estágio tecnológico. Mesmos valores de "idade real"
saída	250000	30000	124930	63674	

Fonte: Dados calculados

Tabela 2-4 – Variáveis dos dados de imóveis (casa)

Variáveis	Máximo	Mínimo	Média	Desv. Padr.	Descrição
bairro	5	2	3.8627	1.0774	Naturalmente um local é "melhor" ou "pior" do que outro em função de diversas características, entre as quais sua infra-estrutura urbana. 1 inferior, 2 razoável, 3 bom, 4 ótimo, 5 centro
garagem	1	0			Identifica a presença de garagem, onde é atribuído o valor mesmo quando há mais que uma vaga.
suíte	1	0			Identifica a presença ou não de suíte.
banheiro	3	1	1.4706	0.6435	Identifica o número de banheiro social.
edícula	1	0			Identifica a presença ou não de edícula.
dist. mercado	3	1	2.2941	0.8317	Identifica a proximidade do imóvel de grandes mercados. 1 mais de 800m, 2 de 500m a 800m, 3 até 500m
área const.	400	70	167.5490	82.4736	Identifica a área total construída.
área terreno	1200	225	525.5686	238.1417	Identifica a área do terreno.
acab.	4	1	2	0.6000	Identifica os vários níveis de acabamento. 1 baixo, 2 normal, 3 alto
cobertura	4	1	3.3235	1.0669	Identifica o tipo de cobertura do imóvel. 1 madeira+eternit, 2 madeira+telha, 3 laje+eternit, 4 laje+telha
estrutura	4	1	2.6078	0.8962	Identifica o material de construção do imóvel. 1 mista, 2 madeira, 3 alvenaria, 4 sobrado.
conserv.	3	1	2.0784	0.7961	Identifica o nível de conservação do imóvel. 1 péssimo, 2 regular, 3 bom, 4 ótimo
piscina	1	0			Identifica a existência ou não de piscina.
dormitório	5	1	2.6275	0.7473	Quantifica o número de dormitórios.
dep.emp.	1	0			Identifica a existência ou não de dependência de empregado, completa ou incompleta. 0 inexistente, 0,5 incompleta, 1 completa.
lav.	1	0			Identifica a existência ou não de lavanderia.
peças	16	5	8.8235	3.0900	Quantifica as peças constituintes do imóvel.
idade aparen.	6	1	3.0980	1.5907	Por ser uma variável continua a idade do imóvel dividiu-se em períodos. 1 vinte anos ou mais, 2 de quinze a vinte anos, 3 de onze a quatorze anos, 4 de seis a dez anos, 5 de dois a cinco anos, 6 até um ano.
saída	290000	15000	96745	58109	

Fonte: Dados calculados

Tabela 2-5 – Variáveis dos dados de imóveis (terreno)

Variáveis	Máximo	Mínimo	Média	Desv Padr.	Descrição
localização	6	2	4.2083	1.3181	Variável que qualifica a localização do imóvel. 1 péssimo, 2 ruim, 3 regular, 4 aceitável, 5 bom, 6 centro
setor comer.	3	0	0.7500	1.1132	Sabendo que os terrenos localizados em zona de comércio ou de moradia, o terreno é mais ou menos valorizado. Esta variável identifica os vários níveis de localização. 0 moradia, 1 bairro, 2 periferia comercial, 3 centro
pólo	1	0			Indica se o imóvel localiza-se próximo a locais que influenciam no seu valor. -1 pólo desvalorizante, 0 inexistente, 1 pólo valorizante
frente	3	1	2.1250	0.8502	Identifica a largura do terreno. Sabendo que um terreno de frente com maior metragem possui uma melhor valorização. 1 até 10m, 2 de 10m a 15m, 3 mais de 15m
área do ter.	2000	242	607.5417	383.4375	Quantifica a área do terreno.
proteção	3	0	1.6667	1.4646	Indica se o terreno possui ou não proteção (muro ou cerca).
plano	3	0	1.9583	0.9991	Identifica se o terreno está acima, abaixo ou ao nível da rua.
inclinado	1	0			Indica o nível de inclinação do terreno.
posição	2	1	1.3333	0.4815	Identifica a posição do terreno na quadra. 1 meio, 2 esquina
pavimentação	1	0			Identifica a presença ou não de pavimentação na rua onde está inserido o terreno.
saída	300000	13000	74458	72403	

Fonte: Dados calculados

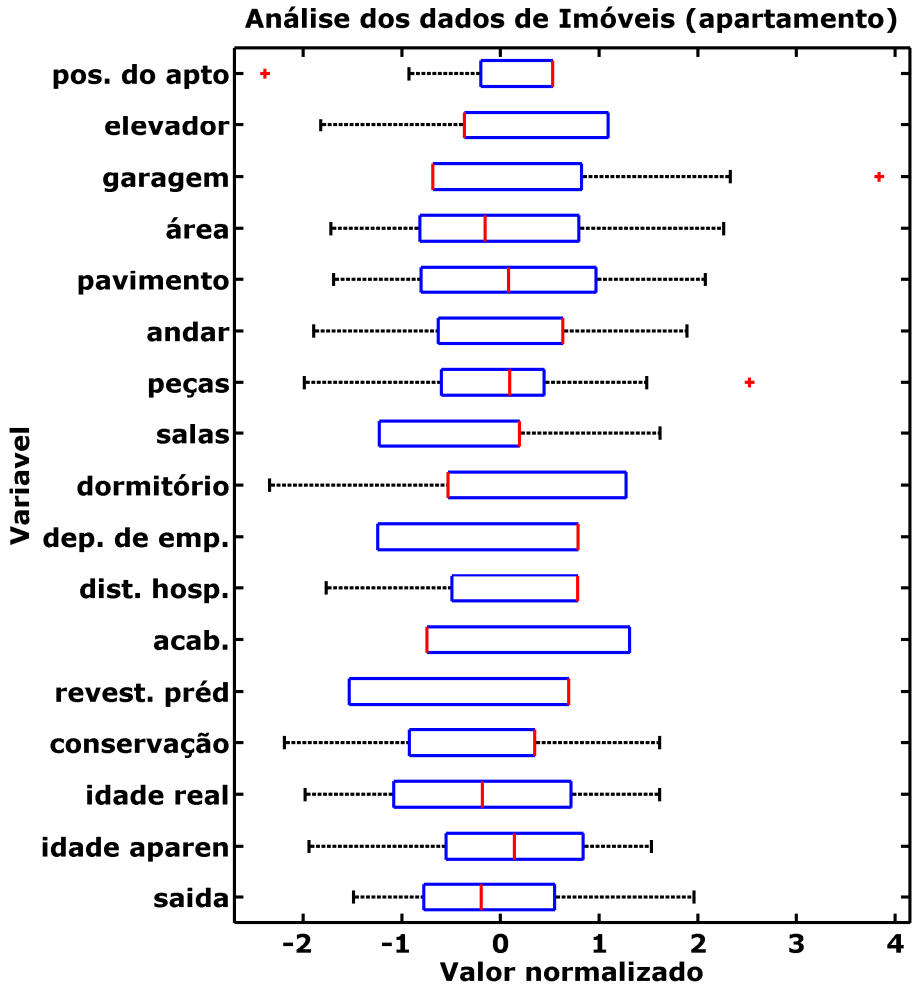

Figura 2-3 – Boxplot com os dados de imóveis normalizados (apartamento)

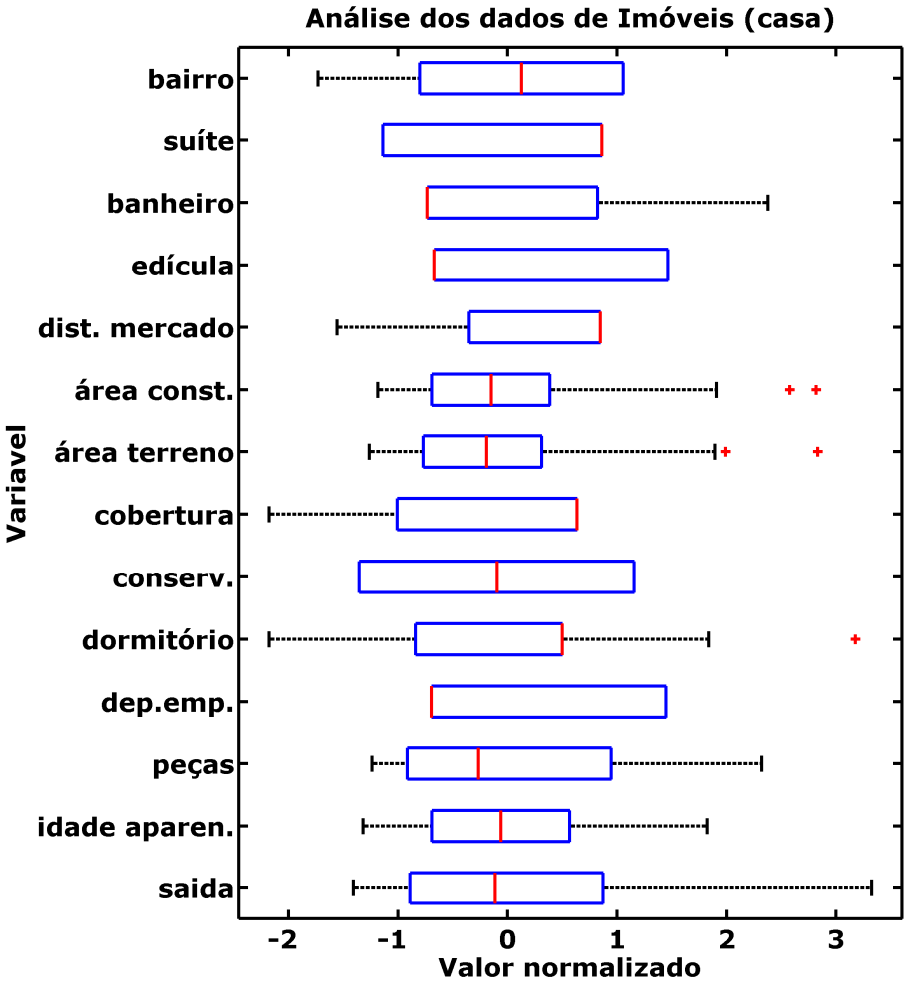

Figura 2-4 – Boxplot com os dados de imóveis normalizados (casa)

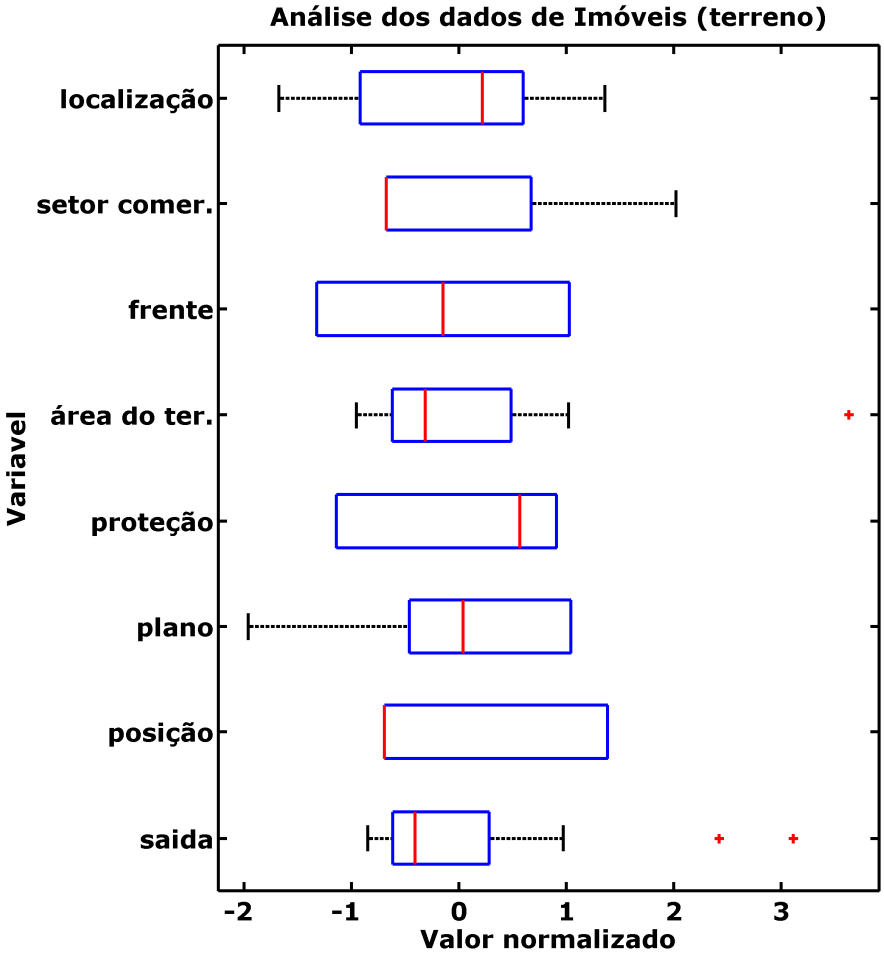

Figura 2-5 – Boxplot com os dados de imóveis normalizados (terreno)

2.4. *Base de dados ABALONE*

Abalone é um tipo de molusco, que vive agarrado em rochas. Sua carne é um prato muito apreciado na Ásia.

O método clássico muito utilizado na determinação da idade de um abalone considera o corte de sua concha em cones, pintando e contando o número de anéis. Porém esse método é muito demorado, logo outras características do abalone foram levantas de um banco de dados com essas características foi criado para tentar encontrar um método mais prático para definir a idade do mesmo.

Esse banco de dados, com 4177 exemplos, contêm as seguintes variáveis:

Tabela 2-6 – Variáveis dos dados de ABALONE

Variáveis	Máximo	Mínimo	Média	Desv. Padr.	Descrição
Male	1	0			Se é macho
Female	1	0			Se é fêmea
Infant	1	0			Se é filhote
Length	0.8150	0.0750	0.5240	0.1201	Comprimento (mm)
Diameter	0.6500	0.0550	0.4079	0.0992	Diâmetro (mm)
Height	1.1300	0	0.1395	0.0418	Altura (mm)
Whole_Weight	2.8255	0.0020	0.8287	0.4904	Peso total (gramas)
Shucked_Weight	1.4880	0.001	0.3594	0.2220	Peso sem casca (gramas)
Viscera_Weight	0.7600	0.0005	0.1806	0.1096	Peso dos órgãos internos (gramas)
Shell_Weight	1.0050	0.0015	0.2388	0.1392	Peso da concha
Saída	29	1	9.9337	3.2242	Número de anéis Idade – 1.5

Fonte: Dados calculados

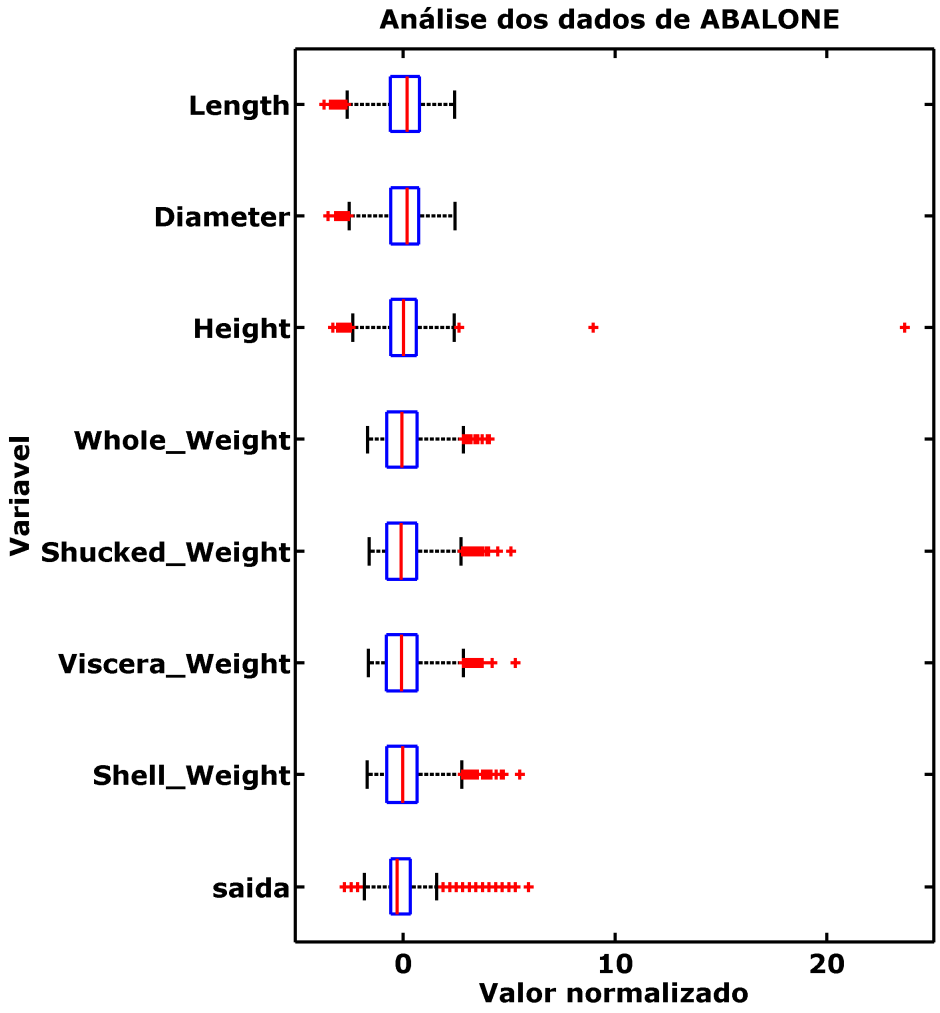

Figura 2-6 – Boxplot com os dados da base ABALONE normalizados

3. Revisão da literatura

O objetivo desse capítulo é apresentar de forma sucinta o conhecimento necessário para o entendimento dos algoritmos utilizados no trabalho. Será apresentado um pequeno histórico sobre cada um deles e como eles funcionam.

São utilizadas quatro técnicas em diferentes partes do problema, AG, RNA, KNN e análise dos componentes principais (PCA).

3.1. Métodos de Regressão

Os métodos de regressão são os métodos com o objetivo de regredir um conjunto de dados a um modelo computacional que irá tentar generalizar a informação contida nesse conjunto de dados.

Aqui iremos revisar dois métodos conhecidos de regressão, RNA e KNN. Os dois métodos utilizam metodologias bem diferentes para tratar o mesmo problema, o que leva a resultados diferentes para um mesmo conjunto de dados.

Nosso trabalho utiliza de duas topologias de RNA e uma formulação do KNN, tentando assim avaliar a melhor metodologia para um conjunto de dados.

Rede Neural Artificial

A RNA é um produto do paradigma do aprendizado supervisionado. Apesar de existirem métodos no estudo de RNA que utilizam de aprendizado não supervisionado este não será utilizado em nosso trabalho.

O aprendizado supervisionado consiste de um processo de indução que procura inferir hipóteses a partir de um conjunto de dados. Torna-se importante o estudo de teorias de generalização no sentido de estabelecer a hipótese ou modelo que seja mais adequada ao problema. Primeiramente, vamos abordar de forma geral o problema de aprendizado supervisionado, em seguida introduziremos a principal teoria de

generalização que utiliza o conceito de aprendizagem estatística de Vapnik e Chervonenkis [20].

Ultimamente, uma classe de problemas ocorrentes na tomada de decisões em um ambiente real, cercado de incertezas e imprecisões, tem sido resolvida, segundo Zadeh [1994], utilizando-se técnicas denominadas *"soft computing"* que empregam, sobretudo, o conceito de aprendizado a partir de dados experimentais ou da experiência do agente com o ambiente no qual se encontra inserido o problema. Estes problemas têm, como aspectos principais, a adaptatividade, a distributividade e, freqüentemente, a não-linearidade. É importante observar que todos estes são características marcantes dos sistemas biológicos naturais.

Procuramos o desenvolvimento de técnicas e algoritmos no sentido de solucionar problemas de identificação, classificação, predição, estimativa e controle de sistemas adaptativos e paramétricos através de um processo contínuo de treinamento, considerando a existência de um conjunto de dados ou informações do ambiente. Este processo cíclico de retro-alimentação possibilita o ajuste progressivo dos parâmetros até que um resultado ótimo seja alcançado.

É importante destacar que o desenvolvimento de modelos adaptativos, ver Figura 3-1, segundo Príncipe, Euliano e Lefebvre [21], é um processo construtivo, a exemplo dos projetos de engenharia, possuindo um conjunto de princípios biológicos, físicos e matemáticos que justificam o seu funcionamento.

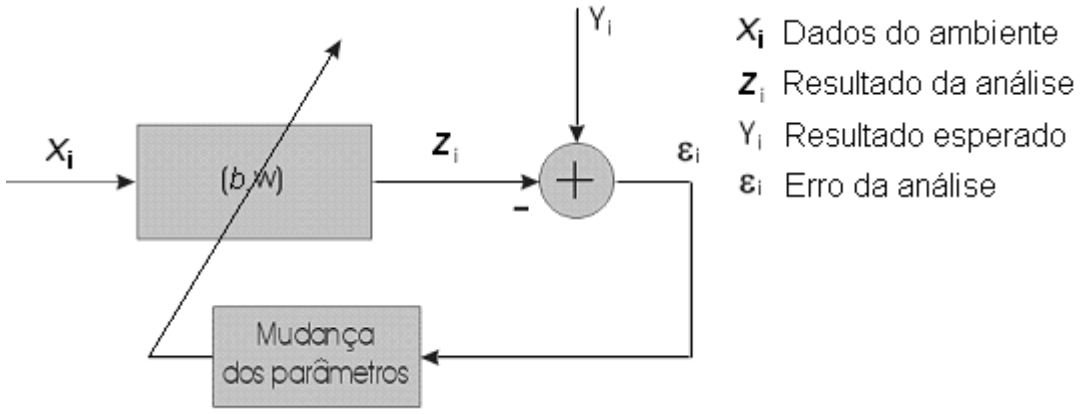

Figura 3-1 – Sistemas adaptativos

Embora muitos problemas desta natureza possuam uma modelagem analítica através de expressões algébricas, equações diferenciais ou sistemas discretos, procuramos uma alternativa de solução mais parecida com a forma humana de atuar. Certamente, a inteligência humana não é capaz de realizar cálculos matemáticos complexos. Entretanto, a sua habilidade em interagir com sistemas adaptativos de alta complexidade é notória e fruto, talvez, de milhões de anos de evolução em contato com a natureza.

Neste sentido, temos um problema fundamental, ou seja, temos que encontrar a forma mais eficiente de se utilizar o conhecimento ou a habilidade humana no desenvolvimento de modelos adaptativos. A princípio, destacam-se duas categorias principais de sistemas. Na primeira abordagem estão os sistemas de arquitetura "cima-baixo" (*top-down)*, onde o conhecimento humano é inserido na forma de heurísticas ou regras de julgamento em modelos simbólicos e declarativos, característicos das técnicas cognitivas de Inteligência Artificial. Apesar deste tipo de arquitetura ter sucesso na solução de inúmeras tarefas de altíssima complexidade, como jogos, planejamento, otimização combinatória e processamento da informação, a mesma é deficiente na incorporação de capacidades mentais como o aprendizado, a discriminação, a

categorização, a generalização e a memorização, ingredientes fundamentais em qualquer sistema inteligente.

Na segunda abordagem, destacam-se os sistemas de arquitetura "baixo-cima" (*botom-up)*, provenientes dos modelos conexionistas de Inteligência Artificial, nos quais o conhecimento e a habilidade humana são conquistados ou aprendidos a partir da utilização de um conjunto de dados experimentais, a partir da interação com o ambiente e, também, a partir de um conjunto de paradigmas de aprendizado que vão direcionar a forma como os dados do modelo serão ajustados.

É nesse conjunto de sistemas que iremos destacar alguns de grande importância. Nosso problema resulta em um ajuste de função, portanto iremos estudar o modelo de RNA conhecido como Adaline, onde temos apenas um neurônio linear, e o modelo utilizado no Perceptron de Multi-Camadas, ou MLP, que adiciona ao modelo Adaline uma camada interna com neurônios não-lineares.

Em ambos os modelos de RNA utilizamos o algoritmo de treinamento "*Levenberg-Marquardt backpropagation*" que proporciona maior velocidade de convergência.

Adaline

O modelo de rede neural artificial conhecido como Adaline [22], veio do trabalho no algoritmo *Least Mean Square* (LMS) [22]. Seu nome ADALINE é devido ao acrônimo – ADAptive LINear Element ou (ADAptive Linear NEuron).

Nele temos apenas um neurônio e este é linear. Ou seja, ele executa um ajuste de função totalmente linear. A Figura 3-2 mostra uma representação desse modelo. Nela temos o vetor X que representa o dado de entrada, o vetor W que representa os pesos do modelo, o scalar B que representa o bias e o scalar Z que representa o resultado obtido pelo modelo.

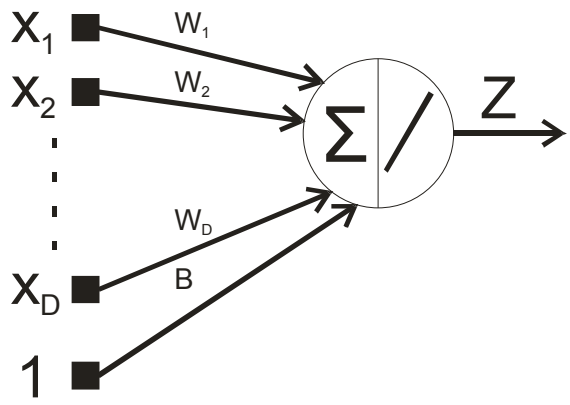

Figura 3-2 – Modelo Adaline

Como pode ser visto esse modelo representa uma regressão linear, que apesar de em sua origem ser treinado pelo algoritmo LMS, em nosso trabalho optamos por utilizar o treinamento "*Levenberg-Marquardt backpropagation*".

Perceptron de múltiplas camadas MLP

O Perceptron é um algoritmo proposto por Rosenblatt [23] que tem como objetivo o reconhecimento de padrões utilizado na classificação dos dados. Sua topologia é muito parecida com o Adaline, porém seu neurônio utiliza uma função não-linear. Foi considerado o primeiro algoritmo de aprendizado relacionado a modelos não-lineares.

Minsky e Papert [24], afirmaram que o poder de aproximação do Perceptron somente seria aumentado se fossem incluídas novas camadas de processamento.

Um Perceptron de multicamadas, ou simplesmente MLP, com uma camada interna, tem sua topologia definida pela Figura 3-3:

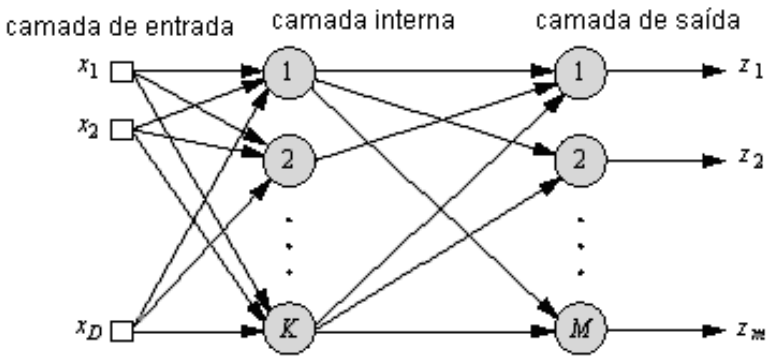

Figura 3-3 – Topologia do Perceptron MLP

Existem alguns resultados interessantes relacionados ao poder de computação das redes MLP, principalmente quando relacionadas ao problema de aproximação de funções.

O primeiro se refere ao fato de que o poder computacional de um Perceptron de múltiplas camadas somente é aumentado se as funções de ativação dos elementos processadores forem não-lineares. De fato, se considerarmos um Perceptron de uma camada interna, ou de duas camadas, como uma composição de funções, ou seja:

$$f(\sum f(\sum(.))) = f\left(\sum f\left(\sum_j w_{ij}x_j + b_i\right)\right) = f(\sum f(net_i)) = f\left(\sum_j w_{ij}.net_j + b_i\right) \quad 3\text{-}1$$

podemos reduzi-lo a um Perceptron de uma única camada de saída se a função de ativação *f* for uma função linear.

O segundo se refere ao fato de que um Perceptron de uma camada interna produz uma região de classificação equivalente à existência de várias regiões convexas abertas ou fechadas, sendo cada uma formada pela interseção de semi-espaços gerados pelos vários hiperplanos definidos pela camada anterior. Neste sentido, o mapeamento de funções na sua forma mais geral, definidas por regiões côncavas, somente é possível em Perceptrons de duas camadas, nos quais a camada de saída será responsável pela combinação de regiões convexas, gerando regiões de classificação côncavas ou não necessariamente convexas.

O terceiro se refere a um teorema derivado do trabalho de Kolmogorov [25]. Este teorema assegura que qualquer função contínua de várias variáveis pode ser representada ou aproximada por um pequeno número de funções de uma variável. Em uma rede MLP isto significa que uma função contínua de dimensão **d** que mapeia valores reais pode ser computada por uma rede de três camadas com **d** unidades na camada de entrada, **d.(2d+1)** unidades na primeira camada interna, **(2d+1)** unidades na segunda camada interna e uma unidade na camada de saída.

O quarto resultado se refere ao poder de classificação das redes de saída sigmóide. Considerando a capacidade de mapeamento ou aproximação universal de uma rede MLP e a utilização de uma função do tipo logística como função de ativação, teremos um classificador Bayesiano.

Aproximação de funções

A aproximação de funções tem como base o teorema da projeção linear, o qual garante que uma função pode ser aproximada por uma combinação linear de uma base de funções elementares ψ_i, associadas a um conjunto de parâmetros, ou seja:

$$\hat{f}(x,a) = \sum a_i \psi_i(x), para\ um\ erro\ \left|f(x) - \hat{f}(x,a)\right| < \xi \qquad 3\text{-}2$$

Karl Weierstrass (1815-1897) provou que polinômios de alguma ordem podem aproximar funções contínuas em determinado intervalo. Formalmente, podemos citar o teorema de Weierstrass como:

"Seja *S[a,b]* o espaço de funções reais e contínuas definidas no intervalo *[a,b]*. Se $f \in S[a,b]$ então existe uma aproximação polinomial para $f(x),\ x \in [a,b]$ na forma:

$$P(x) = \sum_i \alpha_i x_i^i, para\ i = 0, \dots, n \qquad 3\text{-}3$$

para um conjunto de coeficientes α_i reais tal que $|f(x) - P(x)| < \varepsilon$, para $\varepsilon > 0$."

Este teorema foi estendido para a classe de funções sigmóides por Funahashi [26], Cybenko [27] e, posteriormente, para as funções gaussianas, garantindo a ambos os modelos a capacidade de aproximadores universais.

A escolha das funções elementares, considerando a sua forma e quantidade, no sentido de formar uma base de aproximação, deve atender, a princípio, a condição de que as mesmas sejam linearmente independentes, ou seja:

$$\alpha_1\psi_1 + \alpha_2\psi_2 + \cdots + \alpha_n\psi_n = 0\ se\ e\ somente\ se\ \alpha_1 = \alpha_2 = \cdots = \alpha_n = 0 \qquad 3\text{-}4$$

Outra condição imposta ao conjunto de funções elementares, é que a base seja ortonormal, ou seja:

$$\int_a^b \psi_i(x).\psi_j(x)dx = \delta_{i,j}\ onde\ \delta\ representa\ a\ função\ de\ Kronecker \quad 3\text{-}5$$

Isto significa que a projeção ortogonal de um elemento da base em outro elemento da base fornecerá sempre o valor zero, fornecendo um único conjunto de multiplicadores α_i como solução da aproximação ou projeção da função *f*(x). Neste sentido, devemos avaliar o conjunto de multiplicadores através da solução do sistema representado pela equação 3-6:

$$\alpha_i = \langle f(x), \psi_i(x) \rangle \quad 3\text{-}6$$

Pelo fato da base ser ortonormal, temos:

$$f(x) = \sum_i \alpha_i.\psi_i(x) \quad 3\text{-}7$$

Tomando o produto interno entre f(x) e $\psi_1(x)$, computamos:

$$\langle f(x), \psi_1 \rangle = \sum_i \alpha_i.\langle \psi_i(x), \psi_1(x) \rangle = \alpha_1.\langle \psi_1, \psi_1 \rangle \quad 3\text{-}8$$

fornecendo:

$$\alpha_1 = \langle f(x), \psi_1(x) \rangle \quad 3\text{-}9$$

De forma geral, para a avaliação de todos multiplicadores, temos:

$\alpha_i = \langle f(x), \psi_i(x) \rangle$, para a aproximação discreta e

$\alpha_i = \int f(x).\psi_i(x)\,dx$, para a aproximação contínua.

MLP e ajuste de função

Podemos considerar o modelo Perceptron, com uma camada interna com funções de ativação sigmóides e um combinador linear na camada de saída, como uma base de funções elementares, conforme a Figura 3-4. Na Figura 3-4 temos o velor X que representa os dados de entrada, o vetor W que representa os pesos do modelo, o vetor B

que representa os valores dos bias do modelo e o scalar Z que representa o valor de saída do modelo. Neste caso, cada função elementar tem a forma de uma função logística, cujo parâmetro representa uma combinação linear do vetor **x** de entrada acrescido de um bias/viés, ou seja, definimos:

$$\psi_i(x) = \frac{1}{\left(1 + exp(net_i)\right)} \qquad 3\text{-}10$$

Onde:

$$net_i = \sum_j w_{i.j} x_j + b_i \qquad 3\text{-}11$$

Determinando, como saída do modelo, o valor correspondente ao valor da função:

$$z(x) = \sum_i \alpha_i \psi_i(x) \qquad 3\text{-}12$$

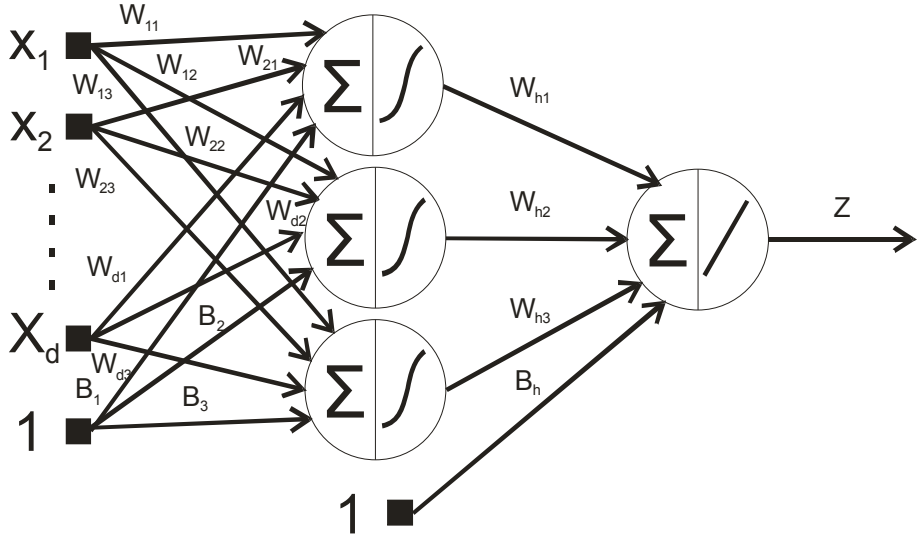

Figura 3-4 – MLP com base de funções

Neste exemplo, relacionado a um Perceptron MLP, consideramos as funções elementares como aproximadores globais, no sentido de que suas entradas respondem ao espaço global do problema. Também, não são estáticas, mas sim adaptativas, dependendo dos parâmetros associados ao processamento da camada interna. Neste sentido, podemos dizer que um Perceptron de uma camada interna, ou uma MLP,

aproxima uma função arbitrária, real e contínua, decidindo a orientação, localização e amplitude de um conjunto de funções de ativação sigmóides multidimensionais, conforme a Figura 3-5. Temos então que a soma das logísticas (linhas mais finas) resultam na função aproximada (linha mais escura).

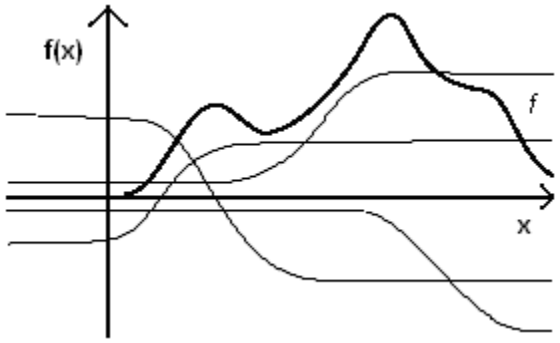

Figura 3-5 – Aproximação de funções com bases logísticas [21]

É essa característica do MLP que iremos utilizar em nosso trabalho.

Backpropagation

O treinamento da rede neural artificial é um procedimento que tem como objetivo ajustar ou adaptar o valor de seus pesos convergindo assim o resultado final da rede para o valor desejado.

O algoritmo *backpropagation*, é o procedimento mais conhecido e utilizado. Foi apresentado pela primeira vez por Paul Werbos em 1974 [28], porém somente depois do trabalho de David E. Rumelhart, Geoffrey E. Hinton e Ronald J. Williams em 1986 [29] ganhou fama e renovou o campo de pesquisa de redes neurais artificiais.

Ele é um algoritmo supervisionado, que utiliza o resultado esperado para adaptar seu modelo. Ele requer que a função de ativação dos neurônios da rede seja diferenciável, por isso muitas vezes utiliza-se a função logística ou tangente.

O treinamento pode ser em linha ou em lote. O treinamento em linha atualiza os pesos cada vez que um dado é inserido no modelo. O treinamento em lote atualiza os pesos em épocas. Consideramos uma época quando todos os dados forem inseridos no modelo.

Seu fluxo na prática é bem simples e pode ser demonstrado no resumo a seguir:

1 - Inicialização: Inicialize os pesos e os bias aleatoriamente, com valores no intervalo [-1; 1];

Definimos o grupo de treinamento: $T = \{x(n),\ d(n)\}$ onde $x(n)$ é a entrada e $d(n)$ a saída desejada.

2 – Apresentação dos dados de treinamento: em linha, executar o passo 3 e 4 para cada indivíduo. Em lote executar o passo 3 e 4 para cada época.

3 – Propagação: Calcular o valor de ativação pela equação 3-13 e o cálculo da saída pela equação 3-14 utilizando a função logística.

$$v_j = \sum_{i=1}^{m} w_{j,i} x_i + b \qquad 3\text{-}13$$

$$f(v) = \frac{1}{1 + e^{-av}} \qquad 3\text{-}14$$

A saída da última camada será a da rede.

4 – Sinal de Erro: Calcular o erro.

$$e_j(n) = d_j(n) - O_j(n) \qquad 3\text{-}15$$

Onde $O_j(n)$ é a resposta da rede.

5 – Retropropagação: Calcular os erros locais desde a saída até a entrada. O gradiente local é definido pela equação 3-16 para a camada de saída ou pela equação 3-17 para as camadas internas.

$$\delta_j(n) = e_j(n) O_j(n) \left(1 - O_j(n)\right) \qquad 3\text{-}16$$

$$\delta_j(n) = e_j(n) \left(1 - O_j(n)\right) \sum \delta_k w_{j,k} \qquad 3\text{-}17$$

Onde: $O_j(n) \left(1 - O_j(n)\right)$ é a função de logística diferenciada, δ_k é o erro das camadas anteriores ligadas à camada j e $w_{j,k}$ os pesos das conexões da camada anterior.

Agora para ajustar os pesos temos:

$$\Delta w_{k,j}(n+1) = \alpha w_{k,j}(n) + \eta \delta_j y_j \quad 3\text{-}18$$

$$w(n+1) = w(n) + \Delta w_{k,j}(n) \quad 3\text{-}19$$

onde α é a constante de momentun, η é a taxa de aprendizagem, δ_j o erro da unidade e y_j a saída produzida pela unidade.

6 – Iteração: O algoritmo repete os passos 3, 4, 5 até que o critério de parada seja satisfeito, como quando o erro está abaixo do definido ou quando o número de iterações chegou ao fim.

Levenberg-Marquardt backpropagation

O algoritmo denominado Levenberg-Marquardt foi inicialmente publicado por Kenneth Levenberg em 1944 [30] e depois redescoberto por Donald Marquardt em 1963 [31]. Ele é uma derivação do método Gauss-Newton [32] que por sua vez é uma variante do método de Newton [33].

O algoritmo faz uso da informação de segunda derivada, ou seja, ele utiliza da matriz Hessiana. Porém encontrar a matriz Hessiana muitas vezes é um problema complexo. Para contorná-lo, da mesma forma que é feito no algoritmo Gauss-Newton, utiliza-se uma aproximação.

Com a informação da Hessiana o método consegue uma convergência mais rápida, finalizando o processo de treinamento da rede em poucas épocas.

Para explicarmos o algoritmo vamos definir a função $J(w)$, na equação 3-20, que representa a soma do erro quadrático de todos os indivíduos de grupo de treinamento do peso w.

$$J(w) = \sum_{i=1}^{N} e_i^2(w) = e^t(w)e(w) \quad 3\text{-}20$$

onde N é o número de indivíduos no grupo de treinamento e $e_i(w)$ é a função de erro do indivíduo *i* para o peso informado *w*.

O gradiente de J pode ser facilmente calculado em:

$$\nabla J(w) = 2\mathcal{J}^T(w)e(w) \quad 3\text{-}21$$

onde $\mathcal{J}$ é a matriz Jacobiana dada por pela equação 3-22.

$$\mathcal{J}(w) = \begin{bmatrix} \frac{\partial e_1(w)}{\partial w_1} & \cdots & \frac{\partial e_1(w)}{\partial w_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial e_N(w)}{\partial w_1} & \cdots & \frac{\partial e_N(w)}{\partial w_n} \end{bmatrix} \quad 3\text{-}22$$

Dessa forma, para obtermos a matriz Hessiana continuamos o processo:

$$\nabla^2 \mathrm{J}(\mathrm{w}) = 2\mathcal{J}^T(w)\mathcal{J}(w) + 2S(w) \quad \text{onde} \quad S(w) = \sum_{i=1}^{N} e(w)\nabla^2 e_i(w) \quad 3\text{-}23$$

Assumindo que $S(w)$ é pequeno, temos nossa aproximação da Hessiana dada por:

$$\nabla^2 \mathrm{J}(\mathrm{w}) \cong 2\mathcal{J}^T(w)\mathcal{J}(w) \quad 3\text{-}24$$

Dessa forma temos uma nova direção e um novo valor para o ajuste de cada peso. Substituindo esse novo ajuste em 3-19 temos:

$$w(n+1) = w(n) - \left[\mathcal{J}^T\big(w(n)\big)\mathcal{J}\big(w(n)\big)\right]^{-1}\mathcal{J}^T\big(w(n)\big)e\big(w(n)\big) \quad 3\text{-}25$$

Ou seja, a nova atualização faz uso da informação de segunda derivada, mas não precisa efetivamente calculá-la. Mas temos agora um novo problema: não se pode comprovar a existência da inversa da Hessiana.

Para contornar essa situação o método Levenberg-Marquardt propõe a soma de parcela λI na Hessiana, onde λ é um escalar denominado parâmetro de amortecimento e I é a matriz identidade.

$$w(n+1) = w(n) - \left[\mathcal{J}^T\big(w(n)\big)\mathcal{J}\big(w(n)\big) + \lambda(n)I\right]^{-1}\mathcal{J}^T\big(w(n)\big)e\big(w(n)\big) \quad 3\text{-}26$$

Além de facilitar o cálculo da inversa da Hessiana, o parâmetro de amortecimento ainda cria uma característica interessante no algoritmo. Ele pode ser um valor que se adapta ao processo de treinamento, tornando o ajuste dos pesos dinâmico.

Para valores grandes de λ tornamos o processo parecido com a correção de descida do gradiente, com passo $\frac{1}{\lambda(n)}$. Para valores pequenos a informação da Hessiana é utilizada quase que totalmente.

Dessa forma, o valor de λ sendo alterado durante o treinamento tende a tornar-se pequeno quando estamos próximo ao ponto de ótimo, onde o algoritmo tem sua melhor taxa de convergência.

Como pode ser visto também, devido à natureza do algoritmo, o mesmo só pode ser executado em lote.

Critério de parada

O processo de treinamento de uma rede neural artificial requer um critério para ser interrompido ou finalizado.

Esse critério influencia no resultado final, pois não se deve treinar demais. Uma rede com *overfitting* representa bem os dados do grupo de treinamento, mas não representa bem os dados do grupo de teste.

Isso ocorre porque quanto maior o tempo de treinamento, maior é o ajuste do modelo ao conjunto de indivíduos do grupo de treinamento, tornando assim o modelo pouco geral. E ao utilizar o mesmo modelo num outro conjunto de indivíduos, como no conjunto de teste, o erro acaba sendo bem maior que o esperado.

Para tentar amenizar esse problema, um conjunto de dados, diferente dos dados de treinamento, é testado a cada época, e o valor do erro neste conjunto de dados é utilizado para definir o critério de parada. Esse conjunto de dados é denominado conjunto de validação cruzada.

Normalmente, após um número de épocas em que o erro do grupo de validação piora, o treinamento da rede é interrompido para tentar manter a generalização do modelo.

Outros critérios podem ser utilizados junto com a validação cruzada. Critérios mais simples como número de épocas ou tempo de treinamento são comuns.

K – Algoritmo dos vizinhos mais próximos (KNN)

O algoritmo KNN foi publicado em 1968 por Cover e Hart [34] e faz uso do conceito simples de distância para tentar classificar um indivíduo. Tendo um conjunto de dados previamente classificados, prever a classificação de um novo indivíduo passa a ser apenas uma questão de avaliar a classificação de seus K vizinhos mais próximos.

Métricas

Primeiro é necessário uma métrica para poder avaliar a distância para seus vizinhos. Existem diversas definições de métrica, as mais conhecidas são:

Euclidiana:

$$D(x,y) = \sqrt{\sum (x_i - y_i)^2} \quad \text{3-27}$$

Quadrado da Euclidiana:

$$D(x,y) = \sum (x_i - y_i)^2 \quad \text{3-28}$$

Manhattan:

$$D(x,y) = \sum |x_i - y_i| \quad \text{3-29}$$

Chebychev:

$$D(x,y) = Max(|x_i - y_i|) \quad \text{3-30}$$

A Figura 3-6 exemplifica a diferença entre as distâncias. A distância do quadrado da Euclidiana é assim utilizada para simplificar seu calculo.

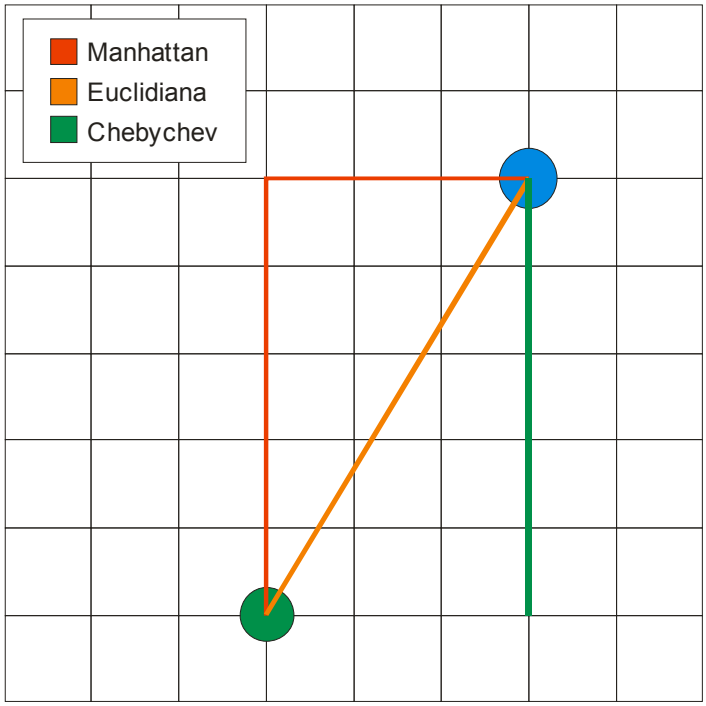

Figura 3-6 – Comparativo entre as métricas: Euclidiana, Manhattan e Chebychev

Classificação

O KNN foi inicialmente criado para problemas de classificação.

O processo é bem simples e pode ser resumido da seguinte maneira: para classificar um novo dado, verificam-se os K vizinhos mais próximos e, ponderando-se pela distância, avalia-se a probabilidade do novo dado ser de mesma classificação que seus vizinhos.

Como no exemplo da Figura 3-7, quando escolhemos 3 vizinhos mais próximos, o novo dado é classificado como triangulo vermelho, quando escolhemos 5 vizinhos a nova classificação passa a ser de quadrado azul.

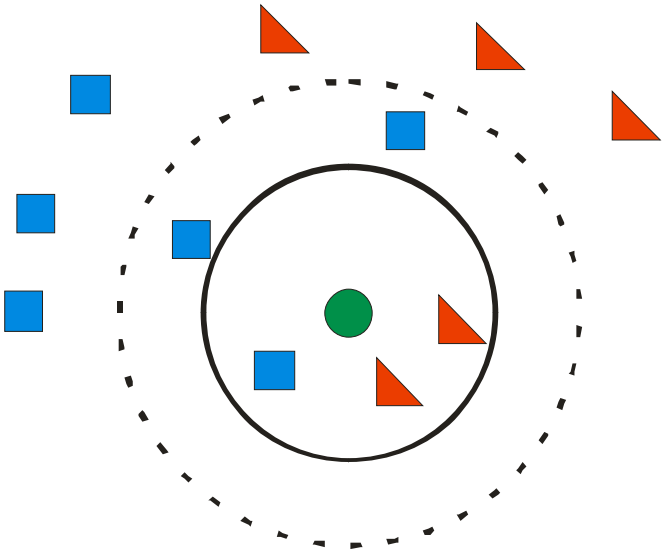

Figura 3-7 – Exemplo do KNN em funcionamento

O algoritmo pode ter diversas variações, como a utilização da distância efetiva na ponderação das classificações e não simplesmente o número de vizinhos, ajudando assim no caso de empate entre uma ou mais classificações.

Aproximação de funções

Outra vertente do KNN é sua utilização no ajuste de funções. A idéia muda um pouco, porém o conceito básico é o mesmo.

Como agora não temos mais classificação, utilizamos os valores de nossos vizinhos para calcular o valor do novo indivíduo. Nesse momento vemos como é necessária a ponderação pela distância, pois é ela que vai ajustar melhor a soma desses valores.

E para fazer esse cálculo utilizamos um conceito de interpolação, uma média ponderada inversamente pela distância. Um dos primeiros estudos nessa área foi realizado por Shepard em 1968 [35], que resultou na equação 3-31, onde p ajusta a ponderação, onde quanto maior o seu valor, maior a importância dos vizinhos mais próximos no valor final. Normalmente o valor utilizado é de $p = 2$.

$$w(x, v_j) = \frac{\frac{1}{D(x, v_j)^p}}{\sum_i \frac{1}{D(x, v_i)^p}} \quad 3\text{-}31$$

Outras definições para o cálculo da interpolação foram publicados, sendo um deles também muito utilizado, o exponencial, visto na equação 3-32, que gera uma curva de ponderação mais suave.

$$w(x, v_j) = \frac{\exp\left(-D(x, v_j)\right)}{\sum_i \exp\left(-D(x, v_i)\right)} \quad 3\text{-}32$$

3.2. Método de Otimização

Para uma melhor utilização dos métodos de regressão é necessários definir bons parâmetros para os mesmos. Esses parâmetros ajustam o funcionamento dos modelos de regressão e esse ajuste pode levar a melhores resultados.

Porém fazer esse ajuste manualmente é muito trabalhoso, por isso em nosso trabalho utilizamos um método de otimização para tentar encontrar os melhores valores para esses parâmetros.

Para isso fizemos um estudo no uso do AG como método de otimização e assim obter um melhor ajuste para nossos dados.

Algoritmo Genético

O AG é uma meta-heurística muito empregada em otimização. Diferente da programação matemática, praticamente todo tipo de problema pode ser acomodado ao AG, ou seja, não existem muitas restrições ao seu uso.

As técnicas empregadas no AG são inspiradas nos estudos de Charles Darwin [4], e seu objetivo é imitar o processo de evolução e tentar assim evoluir a melhor solução para um dado problema.

Nos anos 1950 e 1960, tivemos vários estudos com base nos resultados de Charles Darwin. Muitos problemas na área da ciência da computação [5] e na engenharia [6] foram solucionados com ferramentas utilizando o mesmo principio.

Em 1973 John Holland publicou o primeiro artigo sobre o assunto [7] e em 1975 publicou o livro [8] que teve sua segunda edição publicada no início da década de 90 [9]. Diferente dos demais pesquisadores, Holland queria trabalhar a técnica da evolução de uma maneira mais geral e não focada em um problema específico. E foi dessas pesquisas que surgiu o conceito de Algoritmo Genético.

A mais de 20 anos, diversas pesquisas vem apresentando resultados com o uso e a melhoria do AG. Exemplos disso são: otimização estrutural [10,11,12], otimização de funções multimodais [13] e com restrições [14], processamento de imagem [15], controle de sistemas [16] e inúmeros outros que podem ser encontrados em uma ampla bibliografia.

A idéia é codificar a solução do problema em indivíduos, uma estrutura de dados similar à de um cromossomo, e a partir dos melhores indivíduos gerarmos novas soluções [17]. Isso representa diretamente a teoria da evolução onde os indivíduos mais aptos tendem a se reproduzir mais e com isso passam seu material genético para seus descendentes.

Abaixo estão apresentadas algumas definições importantes para um melhor entendimento do AG [17]:

- Cromossomo: cadeia de caracteres representando informações relativas às variáveis do problema.
- Indivíduo: solução representada por um cromossomo
- Gene: uma unidade básica de um cromossomo
- População: o conjunto de indivíduos ou soluções candidatas
- Geração: Etapa do processo evolutivo da população
- Operações Genéticas: operações realizadas nos cromossomos
- Aptidão: valor numérico do desempenho da solução, utilizado para a seleção de indivíduos para reprodução.

O processo de evolução computacional do AG segue um esquema simples, como demonstrado na Figura 3-8. Existe um grande conjunto de opções que podem ser utilizadas para melhor processo computacional do método.

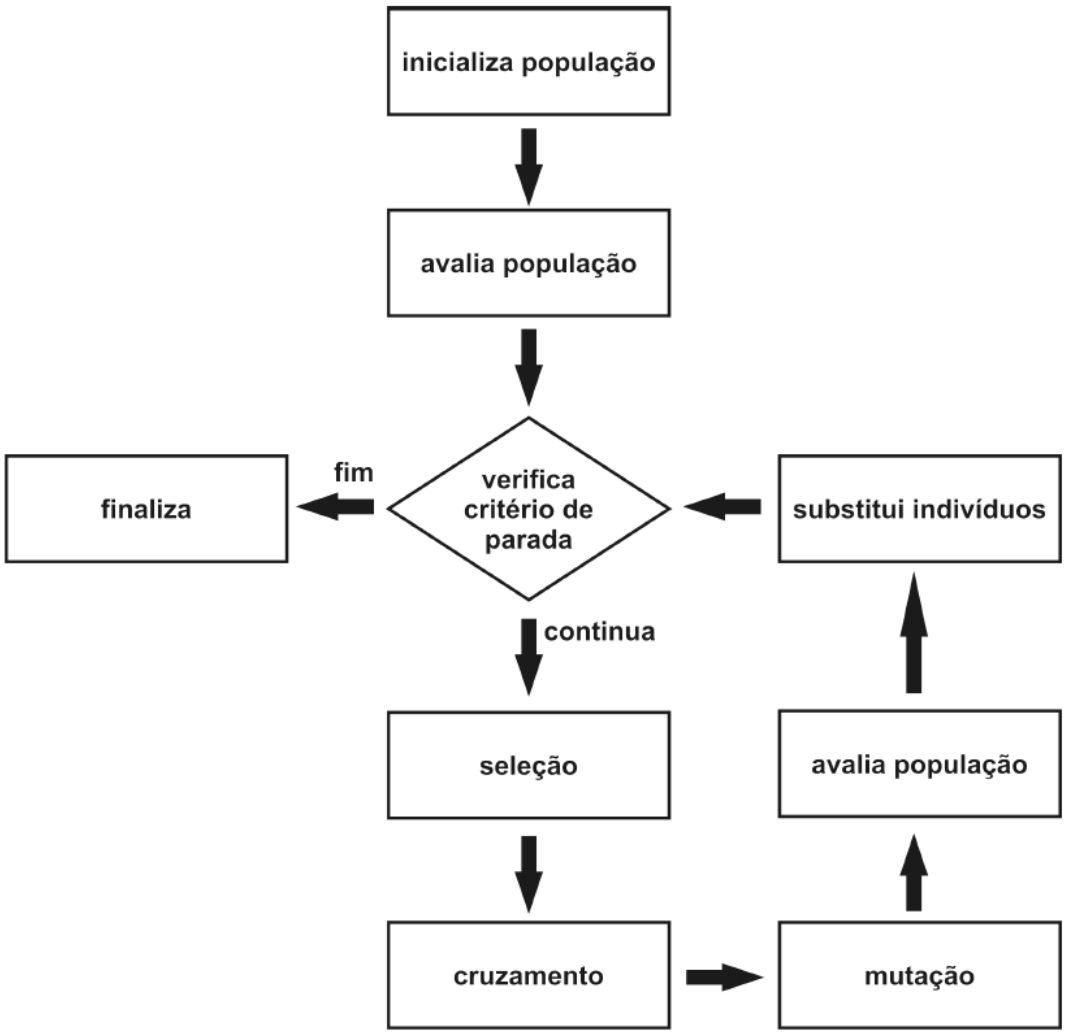

Figura 3-8 – Fluxo de funcionamento de um AG simples

Processo de evolução

Basicamente apenas dois passos da codificação de um AG são dependentes do problema: a codificação dos dados e a função objetivo. A codificação é a metodologia a ser adotada na representação de uma solução candidata do problema em um cromossomo e a função objetivo é a função que irá informar a aptidão do indivíduo, ou seja, ela irá informar quão boa é uma dada solução.

A codificação é uma das mais importantes decisões do processo, pois é a partir dela que iremos representar computacionalmente uma solução candidata do problema [18]. Nele iremos levar as variáveis do problema para um cromossomo artificial. Continuando o paralelo, o gene do cromossomo artificial é a representação individual de cada uma das variáveis do problema.

Essa transformação entre solução/cromossomo é o que diferencia o genótipo do fenótipo. O genótipo é a representação de uma solução em um cromossomo, e o fenótipo representa a informação no contexto do problema. Existem diversas formas de representação do cromossomo. As mais importantes são:

- Binária: utilizada em diversos tipos de problemas, muito utilizado na representação de valores inteiros e contínuos.
- Real: mais utilizada em problemas de variáveis contínuas e muitas vezes também a mais adequada.
- Permutação de símbolos: importante para problemas onde a ordem da informação é importante.
- Árvore: utilizada principalmente em problemas de regressão simbólica. Ex. ajuste de uma expressão algébrica.

A decodificação do cromossomo binário pode ser facilmente exemplificada pela Tabela 3-1:

Tabela 3-1 – Exemplo de codificação

Indivíduo	Cromossomo (Binário)	Fenótipo (Inteiro)
1	100011	35
2	001000	8
3	101011	43
4	111000	56

Fonte: Dados calculados

Diversos estudos são feitos sobre essas codificações, pois ela influencia diretamente na evolução da solução final do problema. Isso poderá ser comprovado mais adiante no trabalho.

Função objetivo

A função objetivo, como já foi dito, leva uma solução ou cromossomo a um valor que será utilizado para avaliar a aptidão do mesmo. Nesta etapa, após a decodificação do cromossomo, as variáveis do problema são repassadas para a função objetivo e a partir do seu resultado podemos inferir sobre a aptidão do indivíduo.

Num exemplo simples podemos utilizar a função $f(x) = x^2$. Utilizando os indivíduos da Tabela 3-1 temos respectivamente os valores de aptidão: 1225, 64, 1849, 3136. Considerando uma otimização para encontrar o ponto de máximo, temos como melhor o indivíduo 4. No caso de encontrar o ponto de mínimo, temos como melhor o indivíduo 2.

Como a função objetivo varia para cada problema, ela pode ser tanto uma simples função matemática, como requerer uma completa simulação computacional. Devido a essa variação no tipo de função objetivo, existem algumas variações no AG para funções mais simples ou mais complexas.

Em nosso trabalho, utilizamos como função objetivo o desempenho de uma RNA e com isso temos a execução de uma simulação, pois a cada execução da função uma nova RNA é treinada.

Processo de seleção

O processo de seleção é o momento de escolhermos, dentre a população atual, os indivíduos que irão propagar seu material genético para a próxima geração.

Existem diversas formas de seleção e cada uma leva em consideração de forma diferente a aptidão do indivíduo. O trabalho de Blickle e Thiele [19] faz um estudo comparativo entre as diversas formas de seleção utilizadas, em especial o seguinte conjunto de processos seletivos: torneio, truncamento, ranking e roleta.

- Roleta: conhecida como seleção proporcional. A probabilidade de um indivíduo ser escolhido é proporcional a sua aptidão.

- Torneio: dois ou mais indivíduos são escolhidos aleatoriamente da população e comparados. O melhor indivíduo é selecionado.
- Truncada: somente uma fração mais apta da população participa da seleção, e neste conjunto todos tem a mesma probabilidade de serem escolhidos.
- Ranking: os indivíduos são ordenados de acordo com sua aptidão e a probabilidade de serem escolhidos é proporcional a esta posição. Essa proporção pode ser linear ou exponencial.

Operadores Genéticos

Em AG temos um conjunto de operadores. São os responsáveis pela manipulação genética dos indivíduos e na geração de novos. É a partir desse processo que os indivíduos passam adiante seu material genético.

Eles podem trabalhar com um ou mais cromossomos. Os operadores de cruzamento ou *crossover* trabalham com mais de um cromossomo, normalmente dois deles. Os operadores de mutação trabalham em apenas um cromossomo por vez.

Operador de Crossover

A operação de *crossover*, ou recombinação, cria um ou mais filhos a partir de seus pais. Normalmente são utilizados dois indivíduos e criados dois novos. Em uma codificação binária costumam ser utilizados *crossover* de um ou dois pontos ou então o *crossover* uniforme.

O *crossover* de um ou dois pontos define posições aleatórias entre os genes dos indivíduos e faz cortes no mesmo. Depois combina cada uma das partes gerando novos indivíduos, como pode ser visto no exemplo das Figura 3-9 e Figura 3-10:

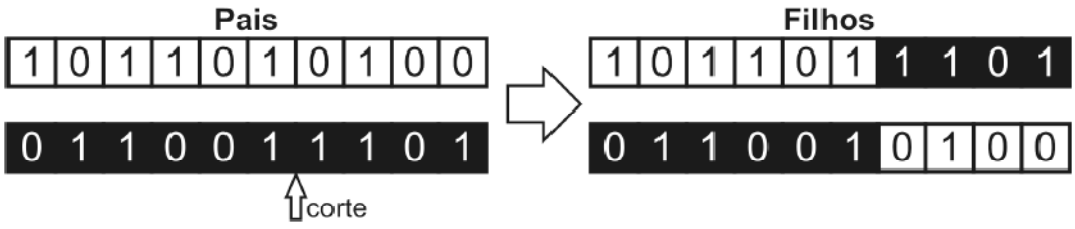

Figura 3-9 – *crossover* de um ponto

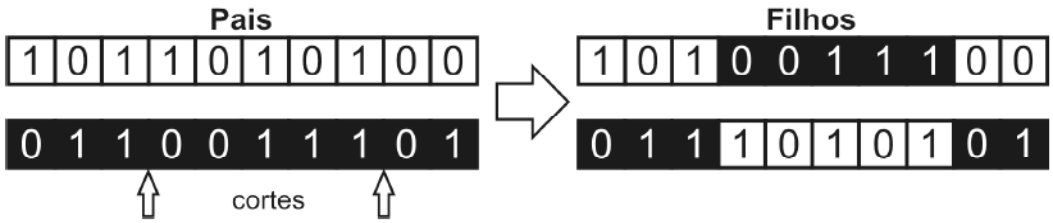

Figura 3-10 – *crossover* de dois pontos

No *crossover* uniforme caminha-se pelo cromossomo e escolhe-se aleatoriamente como será feita a troca de material genético. A Figura 3-11 mostra um exemplo de como o *crossover* uniforme trabalha:

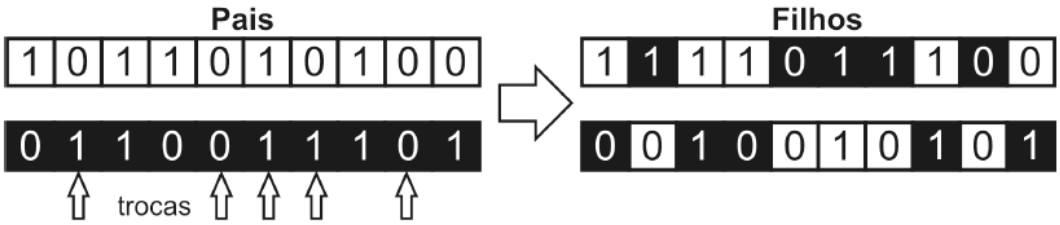

Figura 3-11 – *crossover* uniforme

Não é sempre que os indivíduos se combinam, existe uma probabilidade para que a execução dessa etapa seja feita. Caso o *crossover* não ocorra, é feita uma simples clonagem dos indivíduos selecionados.

Operador de Mutação

Após o processo de combinação, os novos indivíduos passam pelo operador de mutação. Esse operador é muito importante, pois é um operador exploratório e tenta criar uma maior diversidade na população e também combate a convergência prematura.

Ele trabalha de forma simples e com apenas um indivíduo. De acordo com uma probabilidade, operações pontuais são feitas no cromossomo. Como exemplo em uma codificação binária, passamos por cada gene do cromossomo e com certa probabilidade trocamos seu valor. Esse processo é demonstrado na Figura 3-12:

Figura 3-12 – operador de mutação

Outros métodos também podem se utilizados, vai depender da estrutura utilizada. Como também a probabilidade de mutação não precisa ser fixa, ela pode mudar de acordo com as gerações. Normalmente quanto mais próximo do final do processo, menos é a intensidade da mutação, pois nessa etapa o AG tenta refinar uma boa solução e não continuar a exploração.

3.3. Métodos para avaliação dos atributos de entrada

Outra forma de obter um melhor ajuste para seus dados, é fazer um estudo nos seus dados de entrada. Existem algumas maneiras de analisarmos esses dados, utilizamos duas delas. A primeira é uma análise dos atributos de entrada sem levar em consideração os dados de saída, a segunda utiliza da informação de saída tentando assim encontrar os melhores atributos avaliando o resultado obtido.

O método que utilizamos para fazer a primeira análise é o método PCA, ele cria uma melhor organização dos dados possibilitando assim a diminuição dos atributos dos dados de entrada, o que simplifica a regressão dos dados pois diminuímos a complexidade dos métodos que utilizamos para a regressão.

Já para a segunda análise, utilizamos também o AG. Dessa forma, o AG tenta selecionar quais os atributos dos dados de entrada devem participar do processo de regressão de dados para obtermos uma melhor regressão.

PCA – Análise de componentes principais

O método de análise dos componentes principais faz um estudo estatístico sobre os dados de entrada e tenta explicar a estrutura da variância e covariância desses dados através de combinações lineares.

Tem como objetivo evitar o problema de multicolinearidade dos dados criando uma nova representação dos dados com novas variáveis não-correlacionadas. Além disso, as novas variáveis tem a capacidade de acumular uma maior quantidade de informações, possibilitando assim a diminuição do espaço de entrada com apenas uma pequena perda da informação original.

Análise populacional

A PCA tem uma representação algébrica muito simples, pois as novas variáveis são uma combinação linear das variáveis anteriores. Geometricamente a PCA gera uma matriz de mudança de base para os vetores direcionando os novos eixos na direção da maior variabilidade. A Figura 3-13 exemplifica essa mudança de coordenada em dados com duas variáveis.

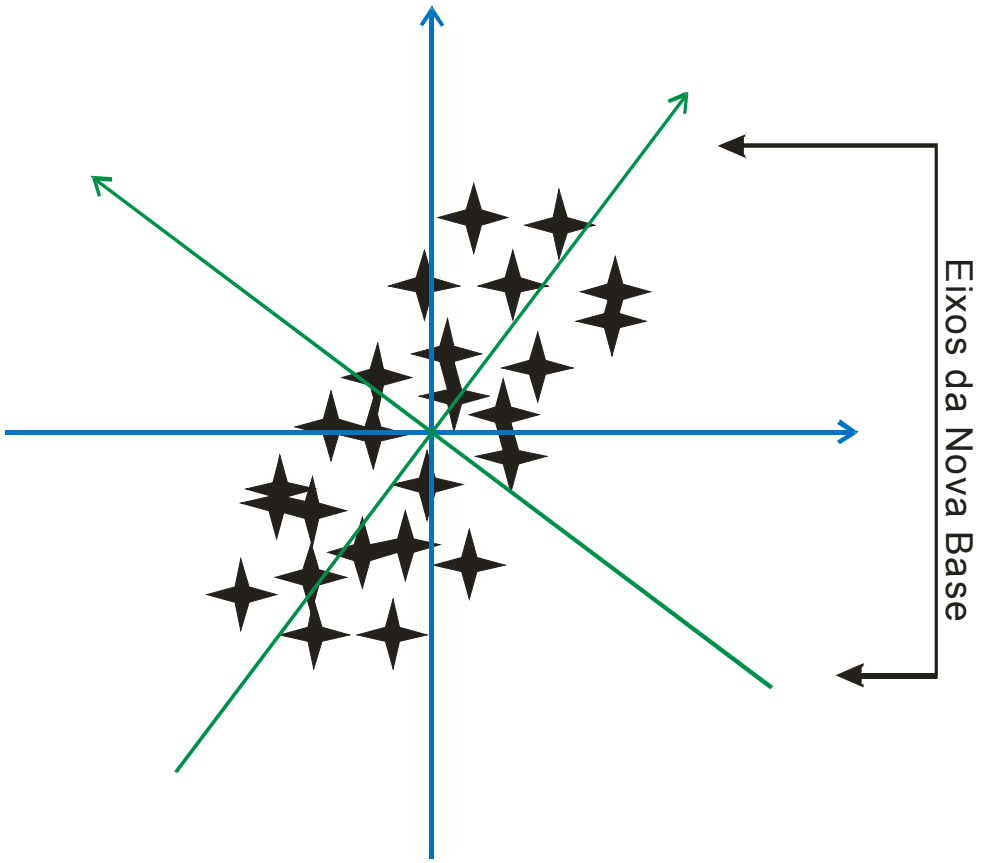

Figura 3-13 – Mudança de base utilizando PCA, os eixos verdes são a nova base

A Figura 3-13 mostra um conjunto de dados em seus eixos originais em azul. Os eixos em verde são os eixos da nova base e se encontram segundo a direção de maior variabilidade. Ou seja, ele avalia em qual dos eixos os dados se espalham mais e ajusta o novo eixo nessa direção, tornando a informação do eixo de menor variabilidade menos importante para descrição da informação.

O calculo da nova base é feito através dos autovalores e autovetores da matriz de covariância $\sum$ ou da matriz de correlação ρ. Os autovetores $e_1, e_2, \dots, e_p$ são ordenados de acordo com seus respectivos autovalores.

Para um melhor entendimento, verificamos a equação 3-33 e definindo assim:

$$V(Y_i) = V(e_i^t X) = e_i^t V(X) e_i = e_i^t \sum e_i \qquad 3\text{-}33$$

- A primeira componente principal é a combinação linear $Y_1 = e_1^t X$ que maximiza a variância de Y_1, sob a restrição de $e_1^t e_1 = 1$

- A segunda componente principal é a combinação linear $Y_2 = e_2^t X$ que maximiza a variância de Y_2, sob a restrição de $e_2^t e_2 = 1$ e $cov(Y_1, Y_2) = 0$
- A i-ésima componente principal é a combinação linear $Y_i = e_i^t X$ que maximiza a variância de Y_i, sob a restrição de $e_i^t e_i = 1$ e $cov(Y_i, Y_k) = 0 \; \forall \; k \neq i$

Análise amostral

Normalmente, quando estamos trabalhando com uma amostra dos dados, não temos a matriz de covariância ∑ ou ρ populacional. Com isso é necessário estimar esses valores a partir da amostra de trabalho.

A matriz de covariância S e a matriz de correlação R são facilmente calculadas a partir dos dados amostrais pelas equações 3-34 e 3-35 respectivamente:

$$S = \frac{1}{m-1}\sum_i (X_i - \bar{X})(X_i - \bar{X}) \qquad 3\text{-}34$$

$$R = D^{-1} S D^{-1} \qquad 3\text{-}35$$

$$D = \begin{bmatrix} S_1 & 0 & \dots & 0 \\ 0 & S_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & S_p \end{bmatrix} \; onde \; S_i \; são \; os \; desvios \; padrões \; das \; variáveis \qquad 3\text{-}36$$

$$\bar{X} = \begin{bmatrix} \overline{X_1} \\ \overline{X_2} \\ \vdots \\ \overline{X_p} \end{bmatrix} onde \; \overline{X_i} \; são \; as \; médias \; das \; variáveis \qquad 3\text{-}37$$

4. Metodologia do Algoritmo

Neste capítulo vamos mostrar o funcionamento, as opções e como foi construído o algoritmo.

Primeiramente todo o algoritmo foi desenvolvido utilizando o sistema MatLab R2007b (7.5.0.342), dessa forma é necessário a instalação do mesmo para execução do algoritmo.

Três funções principais foram criadas: runDescription, runModel, runReport.

4.1. runDescription

Na função runDescription criamos um relatório que exibe os dados do problema. Esse mesmo relatório foi utilizado para a construção do capítulo 2 do trabalho. Nela fazemos uso da toolbox de relatório do MatLab como também da toolbox estatística, onde utilizamos funções de média, desvio padrão e geração do boxplot. Ela é bem simples e auxilia na exploração inicial dos dados ajudando a reconhecer algum dado incorreto, por exemplo.

4.2. runModel

A segunda função, runModel, é a principal do algoritmo. Ela prepara os dados, salva as configurações de execução, gera as amostras, executa o AG em cada um dos problemas e depois monta um gráfico com um comparativo final, salvando os resultados. E é sobre esse funcionamento que iremos tratar agora.

Padronização dos dados

O primeiro passo é padronizar os dados. Tanto os dados de entrada quanto os valores esperados de saída são padronizados. Utilizamos a padronização da distribuição normal, que pode ser vista na equação 4-1.

$$z_i = \frac{x_i - mean(X)}{std(X)} \quad \begin{matrix} onde\ mean\ e\ std\ denotam\ a\ média\ e\ desvio\ padrão, \\ respectivamente\ da\ variável\ x \end{matrix} \qquad 4\text{-}1$$

Guardamos os valores para termos como voltar à escala de valores original.

Essa padronização facilita o trabalho dos métodos computacionais que farão uso dos dados de entrada.

PCA

Verificamos agora, junto às opções de execução, se o algoritmo irá executar a análise dos componentes principais, PCA, reestruturando assim as variáveis de entrada. Caso seja esta a opção, executamos a função *princomp* da toolbox de estatística do MatLab, em sua versão 2.9.2.9. As referências para a teoria utilizada na função são [36, 37, 38, 39].

A utilização do PCA pode simplificar o processo computacional que irá se seguir. Como o PCA reagrupa as informações dos dados de entrada, acumulando um maior número informações nos primeiros atributos, se consideramos a utilização de apenas 98% dessa informação, em determinadas informações os últimos atributos poderão ser deixados de lado, viabilizando assim a utilização de um número menor de atributos.

Esse processo facilita tanto o AG, diminuindo o seu espaço de busca, como nos métodos de regressão, tornando menor a possibilidade da utilização de todos os atributos dos dados de entrada.

Grupos de amostras e suas partições

Seguindo a execução, o próximo passo é a escolha dos grupos de amostras que serão utilizamos no processo de treinamento, validação e teste dos modelos.

São gerados *N* grupos de amostras dos dados de entrada (em nosso trabalho utilizamos 5 grupos), cada um com três partições, uma para treinamento, outra para validação e a terceira para testes. Cada modelo é treinado, validado e testado em cada

dos grupos e no final o erro retorna uma média da partição de teste de cada grupo de amostras.

Ou seja, sempre que formos avaliar um método de regressão e seus parâmetros utilizamos *N* amostras de dados. O método é treinado e validado nas partições de treinamento e validação de cada grupo de amostra e o resultado final é avaliado como uma média dos erros obtidos na partição de teste de cada grupo de amostra.

Isso torna o resultado mais realista, pois se utilizarmos apenas um grupo de amostra podemos obter um falso resultado. Utilizando uma média do valor final de *N* grupos de amostras diminui a probabilidade de um falso resultado.

Nesse processo, também geramos mais *M* outros grupos de amostras (em nosso trabalho utilizamos 30). Esses novos conjuntos de amostras são utilizamos no gráfico de comparação final. A utilização de outros *M* grupos de amostras torna o comparativo ainda mais realista.

Lembrando que durante o processo de treinamento, todas as vezes que um conjunto de atributos e parâmetros são testados, os métodos utilizam os mesmos grupos de amostras, tornando assim a comparação mais justa.

Tanto *N* quanto *M* podem ser informados para a função, como também quantos indivíduos serão utilizados na geração dos grupos de amostra e também qual o percentual desse total irá para cada partição (treinamento, validação e teste). Em nosso trabalho utilizamos um máxima de 500 indivíduos por grupo de amostra e partições com 75%, 5%, 20% respectivamente.

É importante ressaltar que o modelo KNN não utiliza validação cruzada, e com isso em seu treinamento e teste, os indivíduos do grupo de validação cruzada são somados aos do grupo de teste.

RNA (MLP e LMS)

Para os modelos de RNA utilizamos a toolbox do MatLab, versão 5.1.

Criamos a estrutura de cada um dos modelos utilizando a função *network* (v1.9.4.6) e especificamos seus parâmetros de acordo com cromossomo do AG.

O treinamento é feito em cada grupo de amostra com a especificação do percentual de cada grupo.

O critério de parada está relacionado ao número de épocas (um limite de 200 épocas) e ao número de falhas no grupo de validação, que são cinco. Ou seja, o treinamento pára quando alcança 200 épocas ou quando durante o treinamento ele piora o erro no grupo de validação por cinco épocas consecutivas.

Para o treinamento é utilizado a função *trainlm* (v1.1.6.3) que é uma implementação do algoritmo *Levenberg-Marquardt backpropagation*.

O modelo LMS, como foi visto, não utiliza camada interna. E o modelo MLP utiliza *C* camadas internas, de acordo com a informação do cromossomo. Nas camadas internas a função de ativação é uma função tangente.

KNN

O modelo KNN foi uma implementação própria do presente trabalho.

Como foi visto anteriormente é um algoritmo simples: para cada indivíduo do grupo de teste são selecionados os *K* vizinhos mais próximos. Para isso, a distância euclidiana é calculada para cada um dos elementos do grupo de treinamento e os *K* mais próximos são separados.

Desses *K* vizinhos, seus valores de saída são ponderados de acordo com a equação 3-32 e somados, criando assim um valor aproximado para a variável de saída a ser prevista para o novo dado de entrada.

Deve-se ter em mente que a informação de quais atributos/variáveis serão utilizados e de quantos vizinhos serão considerados estão codificados no cromossomo

do AG. Além disso, como o KNN não utiliza um grupo de validação, seus indivíduos são somados aos do grupo de teste.

AG

O algoritmo genético utilizado no trabalho é uma modificação do algoritmo do MatLab. É um AG binário padrão que utiliza operadores de mutação e combinação também padrões.

A toolbox do MatLab (v2.2) contêm boa parte das funções utilizadas no algoritmo. Entretanto, a função que controla os indivíduos selecionados em cada geração e como é feita a substituição do novo individuo selecionado são implementações próprias do presente trabalho.

O algoritmo original seleciona e substitui praticamente toda a população a cada geração, podendo apenas manter os melhores indivíduos (elitismo), caso seja essa a opção desejada.

A modificação introduzida aqui faz com que a cada geração apenas dois indivíduos sejam selecionados da população. O melhor filho gerado é comparado com seus pais e com os indivíduos que perderam no processo de seleção. Caso ele seja melhor que um deles, o mesmo é removido da população e o melhor filho entra em seu lugar.

Dessa forma não existe a opção de elitismo, pois o filho só entra se for melhor, e conseqüentemente não precisamos guardar a melhor solução de toda a evolução, pois ela sempre chega até o final do processo.

O resto do processo acompanha o fluxo padrão do AG. São utilizados operadores de mutação e combinação. O operador de mutação utiliza uma probabilidade baseada em uma distribuição Gaussiana com média em 0 e variância que varia de 1 até 0, onde 1 é o valor na primeira geração e 0 na última geração. Já o operador de

combinação utiliza apenas 1 ponto de corte e é aplicado com uma probabilidade de 80%.

São utilizados *P* indivíduos na população e o AG roda por *G* gerações. Esses parâmetros podem ser modificados de acordo com a escolha do usuário.

Evolução dos modelos

Cada modelo pode participar de *E* execuções do AG, tentando assim, encontrar um novo indivíduo mais apto ou mesmo para confirmar o resultado anterior. Cada execução é considerada individualmente e seu resultado é adicionado no comparativo final.

Para cada modelo o AG evolui um cromossomo de tamanho diferente, devido a diferenças de parâmetros de cada modelo. A única informação em comum de cada modelo é a seleção dos atributos de entrada.

Independente do modelo e dos parâmetros anteriores do algoritmo, bem como a opção de utilizarmos o PCA, o AG tenta encontrar dentre as variáveis de entrada, qual a melhor combinação para o conjunto de dados do problema. Ou seja, dentre as variáveis disponíveis, o AG tenta encontrar quais irão resultar em um modelo com menor erro.

A primeira parte do cromossomo é responsável pela seleção dos atributos que irão participar do processamento. Um gene para cada atributo. Se marcado com 1 o atributo participa, se marcado com 0 não participa.

Os demais genes são responsáveis pelos parâmetros de cada modelo. Os parâmetros do cromossomo de cada modelo são:

- **MLP:** para o MLP o AG evolui as informações de seleção das variáveis, número de neurônios da camada oculta que varia de 1 até 64 (6 posições no cromossomo), taxa de crescimento do parâmetro de amortecimento que varia de 1 até 13,6 (6 posições no cromossomo), taxa de decrescimento do parâmetro de amortecimento que varia de 0,005 até 0,32 (6 posições no cromossomo).

- **LMS:** para o LMS o AG evolui os mesmos parâmetros do MLP, porém como ele não tem camada oculta o parâmetro de número de neurônios não é utilizado no cromossomo.
- **KNN**: para o KNN o AG evolui as informações de seleção das variáveis e o número de vizinhos utilizados no cálculo da regressão que varia de 1 até 8 (3 posições no cromossomo).

Ao final de cada processo de evolução do AG uma imagem com a evolução da média da população e do melhor indivíduo corrente é gravada em disco para uma avaliação do usuário. E, como foi dito anteriormente, aqui treinamos novamente o melhor resultado *M* vezes, utilizando essa informação para o comparativo final.

4.3. runReport

Na função runReport cria-se um relatório final.

Com as informações de todos os indivíduos selecionados por cada execução do AG é gerado um Boxplot final comparando os resultados. Também como índice de comparação é feito um calculo do erro utilizando uma regressão linear simples e uma linha é colocada do gráfico.

Uma pontuação é calculada para definir qual o melhor modelo. Para chegar nessa pontuação, calculamos os valores da mediana, desvio padrão e valor mínimo obtido para cada modelo e ordenamos do pior para o melhor. Somamos a posição de cada um dos modelos em cada uma das ordenações, o resultado dessa soma é a pontuação final.

Ou seja, ordenamos do pior para o melhor os resultados pela mediana. Depois ordenamos do pior para o melhor os resultados por desvio padrão. Depois ordenamos do pior para o melhor os resultados por valor mínimo. Somamos as posições, montando assim uma nova ordenação. Essa nova ordenação que nos entrega um comparativo entre os modelos.

O modelo com maior valor na pontuação é considerado o melhor modelo.

Como todos os dados são colocamos no relatório, o usuário poderá definir outro critério para avaliar o melhor modelo, se assim desejar.

Os parâmetros encontrados pelo AG em cada um dos modelos também são adicionados ao relatório.

Ao final, os dados do melhor treinamento de cada indivíduo serão guardados podendo assim serem exportados e utilizados externamente, de acordo com o interesse do usuário.

Particularmente, no caso da base de dados de complexos proteína-ligante, que motivou o presente trabalho, os parâmetros do modelo final construído (arquitetura e pesos de um RNA, por exemplo) ficam disponíveis para acesso e uso pelo programa de simulação no trabalho subseqüente dentro do fluxo do desenho racional de fármacos baseado em estrutura.

5. Análise dos resultados

Submetemos os bancos de dados já descritos ao nosso algoritmo.

Foram definidos para o treinamento 5 agrupamentos de dados com os percentuais 75%, 5% e 20% para treinamento, validação cruzada e teste respectivamente.

O AG rodou 500 gerações com uma população de 50 indivíduos. Foram feitas 4 rodadas do AG para cada modelo.

Nas bases de dados em que a quantidade de amostras era superior a 500, limitamos o conjunto de treinamento para 500.

Cada banco de dados foi executado duas vezes, uma utilizando o PCA e outra não.

O processo foi executado em um computador com as seguintes configurações:

- Processador AMD Athlon 64 X2 5200+ (2 x 2.6 GHz)
- Placa mãe ASUS M2N-SLI Deluxe
- Memória DDR2 PC2-5300 Kingston 2 x 1Gb
- Sistema Operacional Windows XP SP2

Como resultado são apresentados os valores finais dos erros obtidos, os parâmetros encontrados pelo AG, um *boxsplot* com um comparativo entre os modelos e um gráfico com a evolução do melhor AG.

No gráfico da evolução, temos dois gráficos. O primeiro mostra a evolução da média da população e da melhor solução. O segundo mostra o cromossomo da melhor solução encontrada, onde as barras azuis representam o valor 1 e os espaços vazios o valor 0.

5.1. Dados de Complexo Proteína-Ligante

A função de erro utilizada no calculo dos dados de complexos de proteína ligante foi o MSE, representada da equação 5-1. Utilizamos o MSE por ser a equação mais popular em estudos de regressão.

O tempo de execução do processo foi de 0:56:44 e 0:45:50, sem PCA e com PCA respectivamente.

$$MSE = \frac{1}{2N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \qquad 5\text{-}1$$

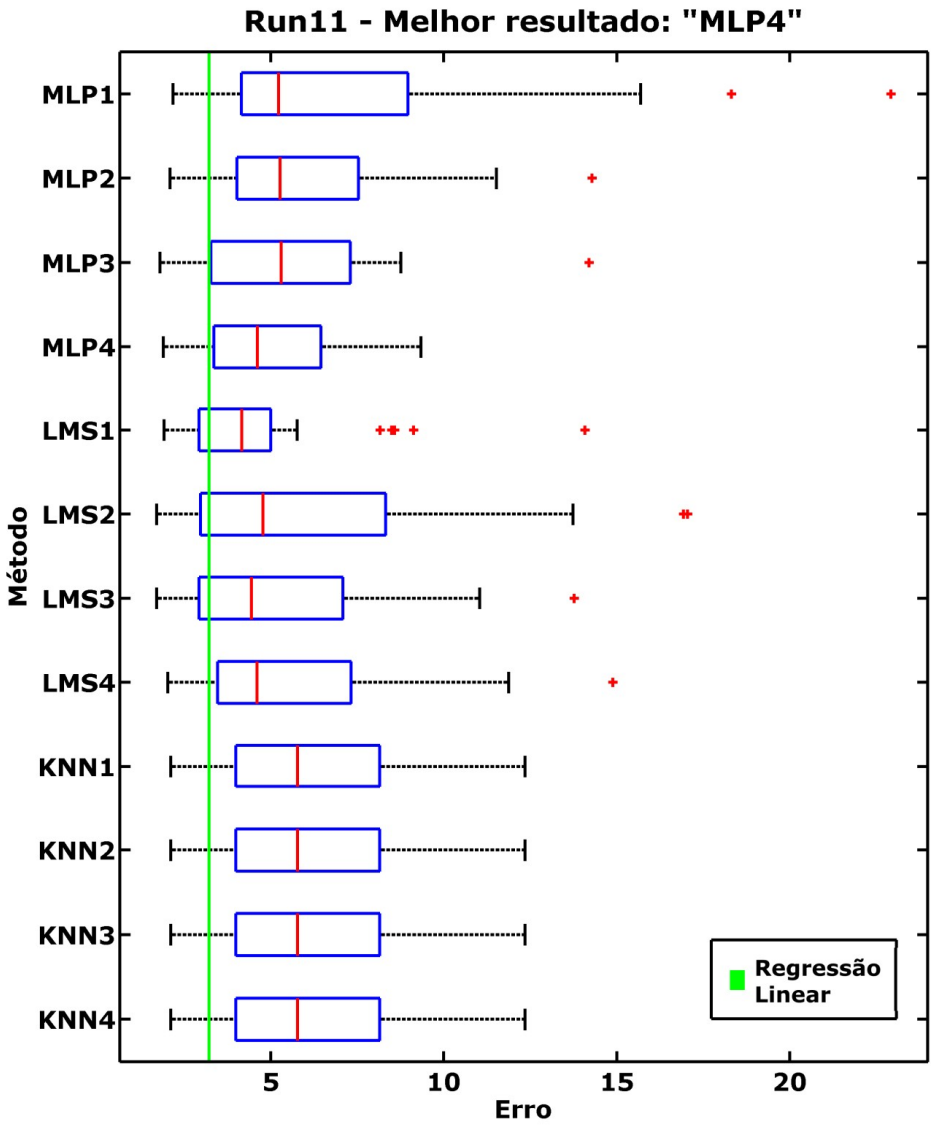

Figura 5-1 – Comparativo dos modelos para complexos proteína-ligante, sem PCA.

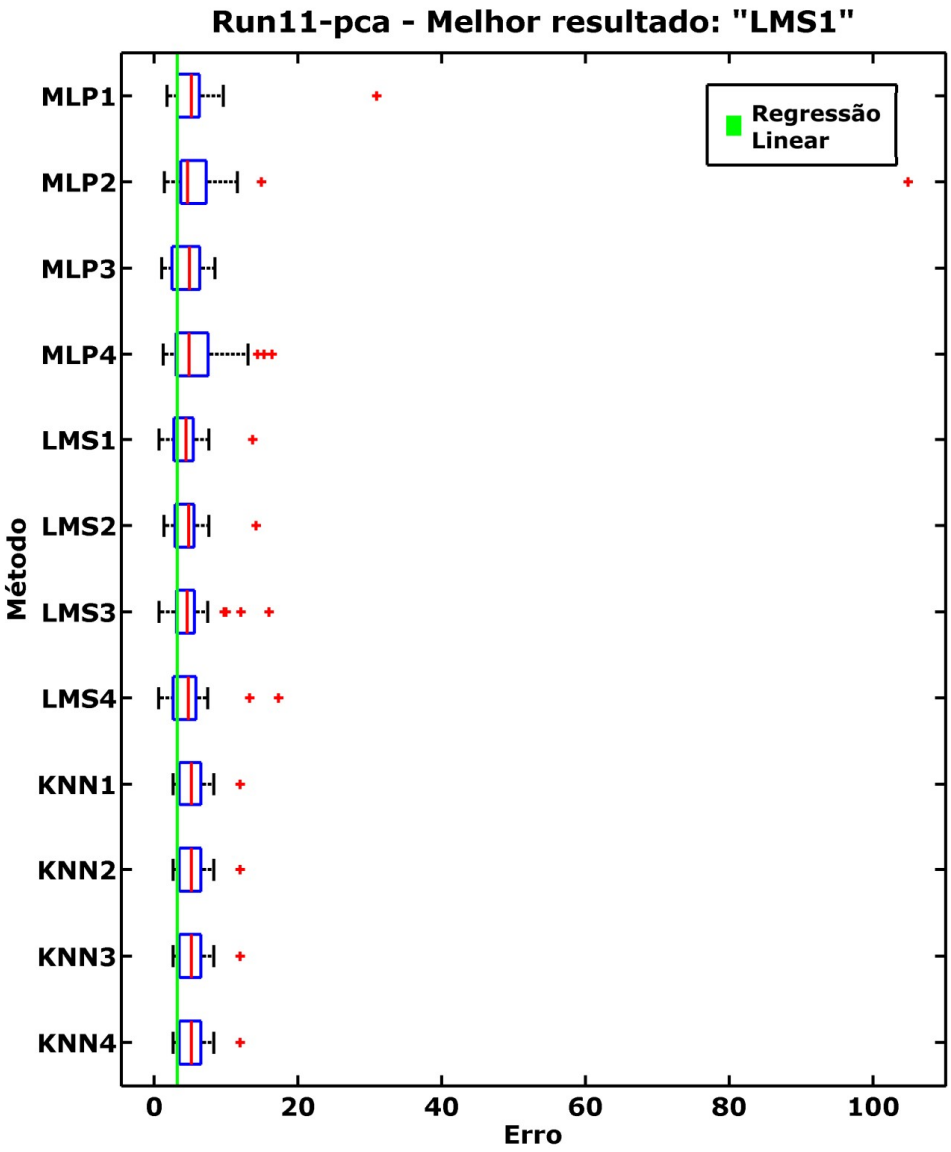

Figura 5-2 – Comparativo dos modelos para complexos proteína-ligante, com PCA.

Tabela 5-1 – Dados estatísticos do resultado dos dados de complexo de proteína-ligante, sem PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	2.1803	4.8854	5.2285	1 + 1 + 7 = 09
MLP2	2.0914	2.8770	5.2783	6 + 5 + 6 = 17
MLP3	1.8060	2.6520	5.3068	10 + 6 + 5 = 21
MLP4	**1.9016**	**2.2142**	**4.6173**	**9 + 12 + 9 = 30**
LMS1	1.9172	2.6384	4.1706	8 + 7 + 12 = 27
LMS2	1.7102	4.2140	4.7745	11 + 2 + 8 = 21
LMS3	1.7101	2.8913	4.4504	12 + 4 + 11 = 27
LMS4	2.0285	3.0789	4.6088	7 + 3 + 10 = 20
KNN1	2.1202	2.5228	5.7822	2 + 8 + 1 = 11
KNN2	2.1202	2.5228	5.7822	3 + 9 + 2 = 14
KNN3	2.1202	2.5228	5.7822	4 + 10 + 3 = 17
KNN4	2.1202	2.5228	5.7822	5 + 11 + 4 = 20

Fonte: Dados calculados

Tabela 5-2 – Dados estatísticos do resultado dos dados complexo de proteína-ligante, com PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	1.8160	5.1545	5.1815	5 +2 +5 = 12
MLP2	1.4586	18.3833	4.6843	6 +1 +10 = 17
MLP3	1.0688	2.0610	4.9601	9 +8 +6 = 23
MLP4	1.2873	4.0731	4.9096	8 +3 +7 = 18
LMS1	**0.7231**	**2.4941**	**4.4668**	**11 +7 +12 = 30**
LMS2	1.4102	2.5333	4.8131	7 +6 +8 = 21
LMS3	0.7231	3.2669	4.5997	10 +5 +11 = 26
LMS4	0.6533	3.4724	4.7925	12 +4 +9 = 25
KNN1	2.6738	2.0543	5.2050	1 +9 +1 = 11
KNN2	2.6738	2.0543	5.2050	2 +10 +2 = 14
KNN3	2.6738	2.0543	5.2050	3 +11 +3 = 17
KNN4	2.6738	2.0543	5.2050	4 +12 +4 = 20

Fonte: Dados calculados

Pelo comparativo no gráfico e pelas tabelas, podemos definir que o LMS1 com o uso do PCA obteve o melhor resultado.

A Figura 5-3 mostra a evolução do AG e seu resultado final. A Tabela 5-3, Tabela 5-4, Tabela 5-5,

Tabela 5-6, Tabela 5-7 e Tabela 5-8 mostram os resultados do AG em todos os modelos.

Tabela 5-3 – Parâmetros encontrados pelo AG – Complexo de proteína-ligante sem PCA - MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	7	0.0350	6.4000	"C", "D", "F"
MLP2	3	0.0150	7.4000	"B", "D", "G"
MLP3	2	0.0100	7.4000	"B", "C", "D", "G"
MLP4	1	0.0050	5.4000	"B", "D", "G"

Fonte: Dados calculados

Tabela 5-4 – Parâmetros encontrados pelo AG - Complexo de proteína-ligante sem PCA - LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.2650	3.2000	"B", "D"
LMS2	0.1400	13.0000	"B", "D", "G"
LMS3	0.3150	4.6000	"B", "D", "G"
LMS4	0.0850	4.4000	"B", "C", "D"

Fonte: Dados calculados

Tabela 5-5 – Parâmetros encontrados pelo AG - Complexo de proteína-ligante sem PCA – KNN

Modelo	K	Variáveis
KNN1	8	"B", "D", "E", "F"
KNN2	5	"B", "D", "E", "F"
KNN3	2	"B", "D", "E", "F"
KNN4	1	"B", "D", "E", "F"

Fonte: Dados calculados

Tabela 5-6 – Parâmetros encontrados pelo AG - Complexo de proteína-ligante com PCA - MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	5	0.0250	12.8000	1,5 - (43.5818%)
MLP2	2	0.0100	12.0000	2,4,5 - (46.7659%)
MLP3	2	0.0100	6.4000	4,5 - (20.2066%)
MLP4	5	0.0250	7.2000	1,5 - (43.5818%)

Fonte: Dados calculados

Tabela 5-7 – Parâmetros encontrados pelo AG - Complexo de proteína-ligante com PCA - LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.1400	3.4000	4,5 - (20.2066%)
LMS2	0.1850	4.2000	1,4,5 - (54.4469%)
LMS3	0.0350	3.8000	4,5 - (20.2066%)
LMS4	0.1500	11.6000	1,4,5 - (54.4469%)

Fonte: Dados calculados

Tabela 5-8 – Parâmetros encontrados pelo AG - Complexo de proteína-ligante com PCA - KNN

Modelo	K	Variáveis
KNN1	6	2,4,5 - (46.7659%)
KNN2	5	2,4,5 - (46.7659%)
KNN3	2	2,4,5 - (46.7659%)
KNN4	4	2,4,5 - (46.7659%)

Fonte: Dados calculados

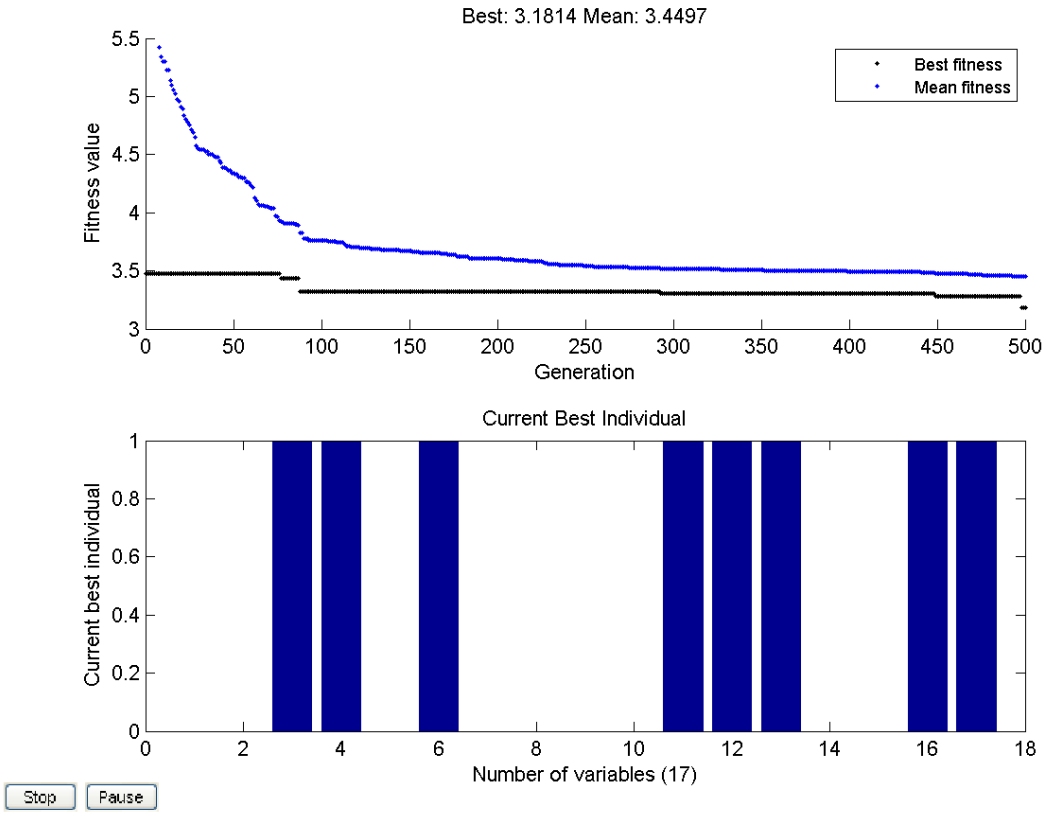

Figura 5-3 – Evolução do AG do LMS1 com PCA e seu resultado final

Avaliação do Resultado

Os dados de complexo de proteína-ligante mostraram uma melhor adequação nos modelos de RNA, como pode ser visto na Figura 5-1 e Figura 5-2.

O uso do PCA ajudou na obtenção de uma melhor regressão, como foi descrito nas tabelas de análise estatística Tabela 5-1 e Tabela 5-2.

Uma observação pertinente é também o fato de nosso melhor resultado utilizar um pouco mais de 20% da informação de entrada, como pode ser vista na Tabela 5-7. O mesmo pode ser observado nos demais resultados onde utilizamos em média apenas 50% da informação de entrada, tanto no uso do PCA quando sem o uso do PCA, onde em média foram escolhidas apenas três variáveis de entrada.

5.2. Dados de Consumo de Energia

A função de erro utilizada no calculo dos dados de consumo de energia foi o MAPE, representada da equação. Utilizamos a função de erro MAPE pois outros trabalhos publicados na área de consumo de energia fizeram uso da mesma.

O tempo de execução do processo foi de 06:20:15 e 04:34:53, sem PCA e com PCA respectivamente.

$$MAPE = 100 * \frac{1}{N} \sum_{i=1}^{N} \left| \frac{\hat{y}_i - y_i}{y_i} \right| \quad \text{5-2}$$

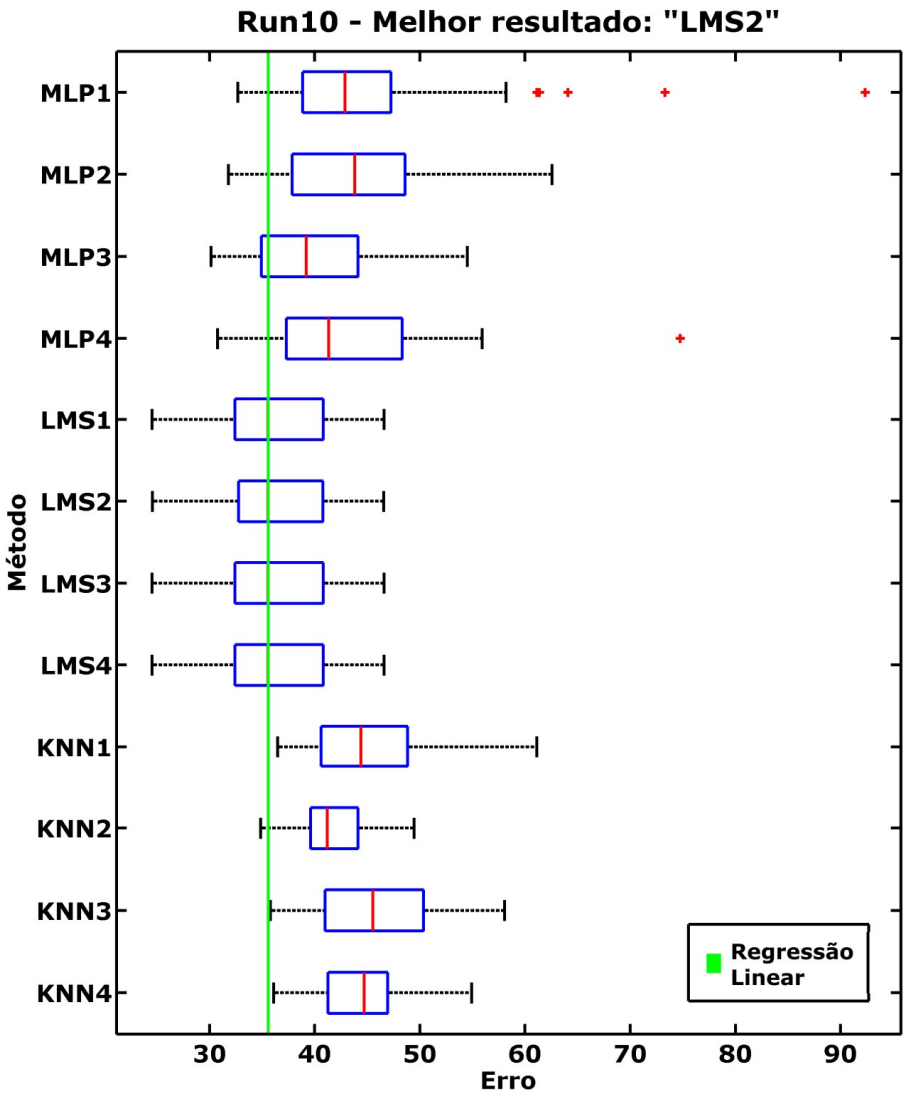

Figura 5-4 – Comparativo dos modelos com dados de Consumo de Energia, sem PCA

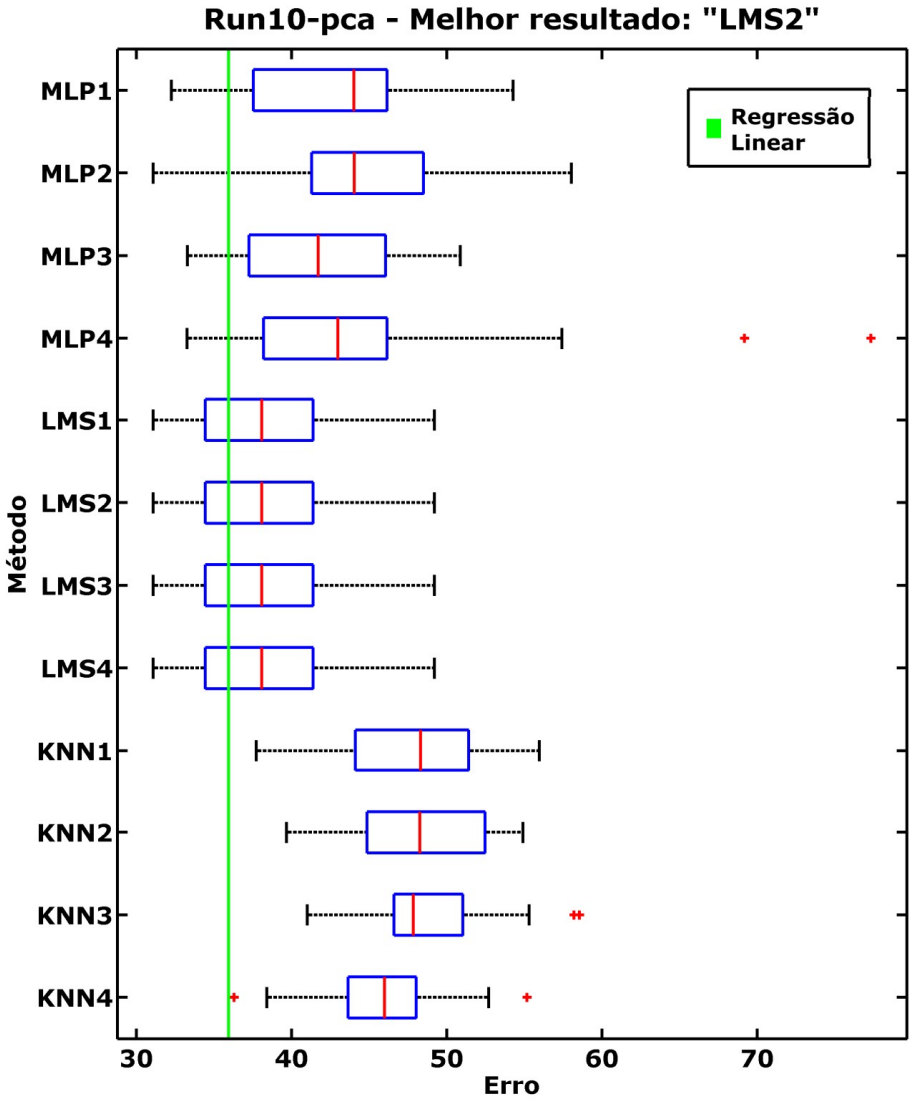

Figura 5-5 – Comparativo dos modelos com dados de Consumo de Energia, com PCA

Tabela 5-9 – Dados estatísticos do resultado dos dados de Consumo de Energia, sem PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	32.6876	12.8193	42.8761	5 +1 +5 = 11
MLP2	31.8036	7.5733	43.8237	6 +3 +4 = 13
MLP3	30.1413	6.3409	39.1906	8 +4 +8 = 20
MLP4	30.7728	8.7423	41.3221	7 +2 +6 = 15
LMS1	24.5469	5.4848	35.5497	12 +8 +9 = 29
LMS2	**24.5757**	**5.4480**	**35.5448**	**9 +10 +12 = 31**
LMS3	24.5470	5.4848	35.5493	10 +7 +11 = 28
LMS4	24.5470	5.4847	35.5495	11 +9 +10 = 30
KNN1	36.4928	5.7480	44.3921	1 +5 +3 = 09
KNN2	34.8842	4.1079	41.2071	4 +12 +7 = 23
KNN3	35.7968	5.5560	45.5459	3 +6 +1 = 10
KNN4	36.1072	4.6758	44.6990	2 +11 +2 = 15

Fonte: Dados calculados

Tabela 5-10 – Dados estatísticos do resultado dos dados de Consumo de Energia, com PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	32.2576	5.5232	44.0155	7 +3 +6 = 16
MLP2	31.0862	6.3443	44.0503	12 +2 +5 = 19
MLP3	33.2836	5.1974	41.7207	5 +4 +8 = 17
MLP4	33.2778	9.5795	42.9966	6 +1 +7 = 14
LMS1	31.0975	4.7046	38.0896	10 +9 +9 = 28
LMS2	**31.0973**	**4.7047**	**38.0894**	11 +6 +12 = 29
LMS3	31.0976	4.7047	38.0894	9 +7 +11 = 27
LMS4	31.0976	4.7046	38.0895	8 +8 +10 = 26
KNN1	37.7274	5.0307	48.3118	3 +5 +1 = 09
KNN2	39.6844	4.5658	48.2687	2 +10 +2 = 14
KNN3	41.0260	3.9862	47.8369	1 +12 +3 = 16
KNN4	36.3011	4.2616	45.9801	4 +11 +4 = 19

Fonte: Dados calculados

Pelo comparativo no gráfico e das tabelas, podemos definir que o LMS2 sem o uso do PCA obteve um bom resultado.

A Figura 5-6 mostra a evolução do AG e seu resultado final. A Tabela 5-11, Tabela 5-12,

Tabela 5-13, Tabela 5-14, Tabela 5-15 e Tabela 5-16 mostram os resultados do AG em todos os modelos.

Tabela 5-11 – Parâmetros encontrados pelo AG - Consumo de Energia sem PCA - MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	22	0.1100	7	"2comp", "3comp", "carros", "tipo3", "area1", "area5", "area6", "banheiro", "relogio", "mono", "bifase", "resident"
MLP2	42	0.2100	3.8	"analfa", "2comp", "area1", "area2", "comodos", "banheiro", "mono", "bifase", "trifase", "resident"
MLP3	27	0.1350	11.4	"1inc", "carros", "comodos", "relogio", "mono", "bifase", "trifase", "resident"
MLP4	19	0.0950	13.2	"empregd", "3inc", "3comp", "carros", "tipo1", "banheiro", "bifase", "resident", "coord_e"

Fonte: Dados calculados

Tabela 5-12 – Parâmetros encontrados pelo AG - Consumo de Energia sem PCA - LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.2000	2.4	"empregd", "3inc", "3comp", "carros", "tipo2", "area2", "comodos", "relogio", "mono", "bifase", "trifase", "resident", "coord_n"
LMS2	0.1800	11	"empregd", "3inc", "3comp", "carros", "tipo1", "area2", "comodos", "relogio", "mono", "trifase", "resident", "coord_n"
LMS3	0.0400	12.6	"empregd", "3inc", "3comp", "carros", "tipo2", "area2", "comodos", "relogio", "mono", "bifase", "resident", "coord_n"
LMS4	0.0650	8	"empregd", "3inc", "3comp", "carros", "tipo2", "area2", "comodos", "relogio", "mono", "bifase", "resident", "coord_n"

Fonte: Dados calculados

Tabela 5-13 – Parâmetros encontrados pelo AG - Consumo de Energia sem PCA – KNN

Modelo	K	Variáveis
KNN1	3	"2inc", "2comp", "3comp", "carros", "area3", "area6", "relogio", "mono", "bifase", "resident"
KNN2	3	"empregd", "analfa", "1inc", "2inc", "2comp", "carros", "tipo3", "mono", "bifase", "trifase", "resident"
KNN3	7	"3comp", "carros", "tipo1", "tipo2", "tipo3", "area1", "area5", "area6", "banheiro", "relogio", "mono", "bifase", "trifase", "resident"
KNN4	8	"2inc", "2comp", "3comp", "area1", "area4", "area6", "banheiro", "trifase", "resident"

Fonte: Dados calculados

Tabela 5-14 – Parâmetros encontrados pelo AG - Consumo de Energia com PCA – MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	3	0.0150	6.8000	1,3,5,10,11,12,16,17 - (43.9492%)
MLP2	7	0.0350	13.6000	1,3,8,9,10,11,12,15,16,21 - (49.7518%)
MLP3	1	0.0050	2	1,2,3,5,6,8,9,11,12,14,16,17,18,19,20,21 - (75.0748%)
MLP4	1	0.0050	5.4000	1,3,9,11,13,15,16,19,20 - (43.3847%)

Fonte: Dados calculados

Tabela 5-15 – Parâmetros encontrados pelo AG - Consumo de Energia com PCA – LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.0550	1.6000	1,2,3,5,8,10,11,12,13,14,15,16,17,18,19,20 - (74.8168%)
LMS2	0.1750	4.2000	1,2,3,5,8,10,11,12,13,14,15,16,17,18,19,20 - (74.8168%)
LMS3	0.2350	8.6000	1,2,3,5,8,10,11,12,13,14,15,16,17,18,19,20 - (74.8168%)
LMS4	0.1150	13.2000	1,2,3,5,8,10,11,12,13,14,15,16,17,18,19,20 - (74.8168%)

Fonte: Dados calculados

Tabela 5-16 – Parâmetros encontrados pelo AG - Consumo de Energia com PCA – KNN

Modelo	K	Variáveis
KNN1	1	1,5,14,16,17,21 - (30.3878%)
KNN2	3	1,3,10,16,17,18,21 - (35.1751%)
KNN3	2	2,3,11,17,19,20,21 - (28.4478%)
KNN4	7	1,5,13,16,17,21 - (30.5017%)

Fonte: Dados calculados

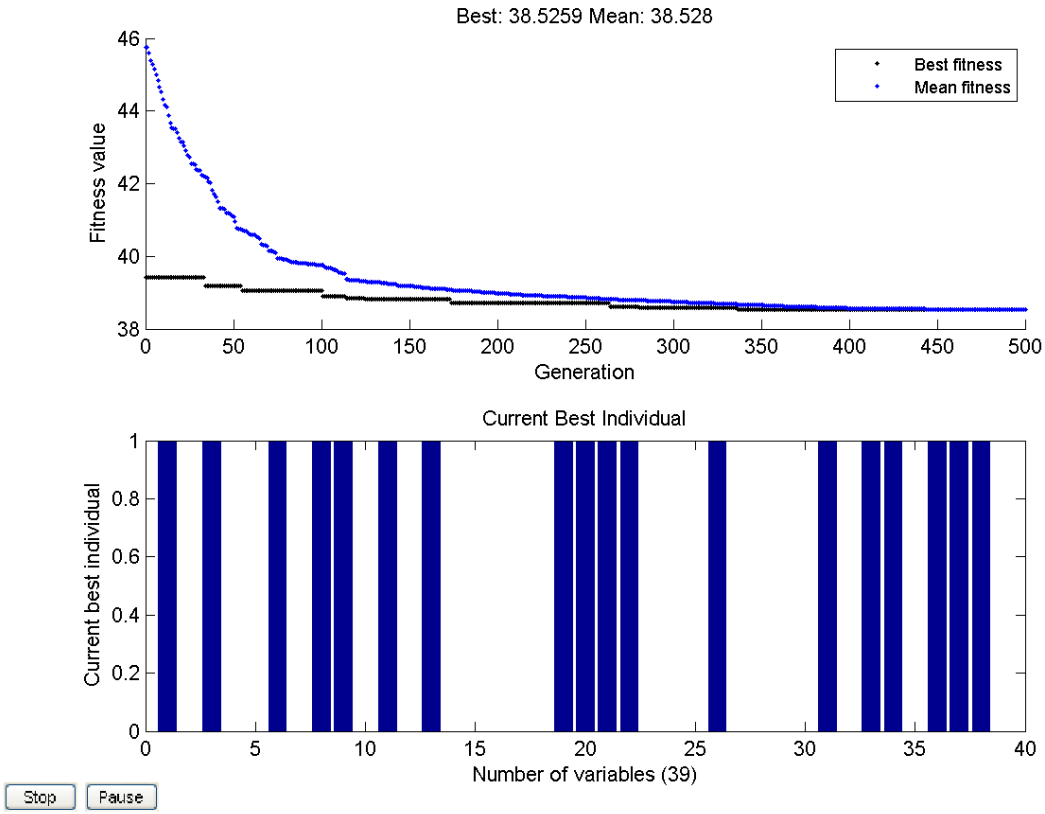

Figura 5-6 – Evolução do AG do LMS2 sem PCA e seu resultado final

Avaliação do Resultado

O banco de dados de consumo de energia se mostrou melhor no modelo de LMS, tanto com valores mínimos menores quanto na dispersão dos dados, como pode ser visto na Tabela 5-9 e na

Tabela 5-10.

Podemos ver também que o PCA não ajudou no ajuste dos dados e em grande parte piorou a regressão.

É interessante observar também a seleção das variáveis: apenas 12 das 27 foram selecionadas. Porém, com o uso do PCA nosso melhor resultado utilizou praticamente 75% da informação de entrada.

Comparação com outros trabalhos

Comparando nosso melhor resultado (LMS2 sem PCA) com o resultado do trabalho apresentado em [40].

Os 444 indivíduos do banco de dados foram divididos nos grupos de treinamento, validação cruzada e teste com 364, 30 e 50 indivíduos respectivamente.

Podemos ver a comparação entre os resultados na Figura 5-7. Como pode ser visto nosso resultado ainda pode melhorar muito em relação ao trabalho anterior. Considerando que no trabalho anterior a metodologia foi semelhante com a de nossa ferramenta, é possível que uma busca maior pelo AG já leve a um resultado compatível.

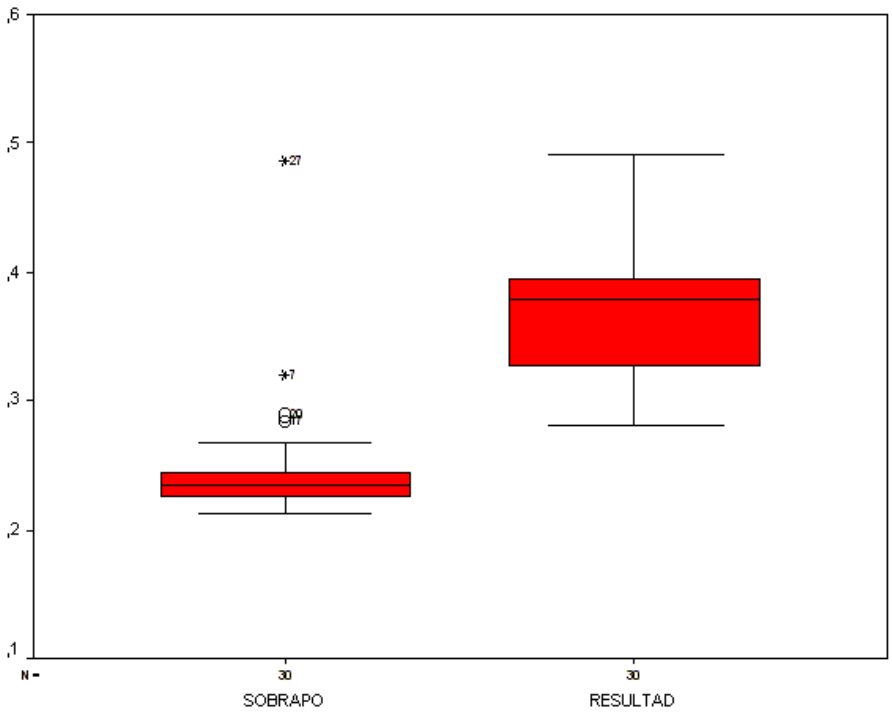

Figura 5-7 – Comparação dos resultados dos dados de Consumo de Energia

5.3. Dados de Preço de Imóveis

A função de erro utilizada no calculo dos dados de imóveis foi o EAX, representada da equação 5-3. Utilizamos a função de erro EAX apenas devido a comparação com o trabalho por Bráulio [2].

$$EAX = \sum_{i=1}^{N} |\hat{y}_i - y_i| \qquad 5\text{-}3$$

Dados de Preço dos Apartamentos

Pelas Figura 5-8 e Figura 5-9 e avaliando a mediana, o valor mínimo e a dispersão, podemos definir que o KNN4 com o uso do PCA obteve um bom resultado.

O tempo de execução do processo foi de 04:36:47 e 01:40:01, sem PCA e com PCA respectivamente.

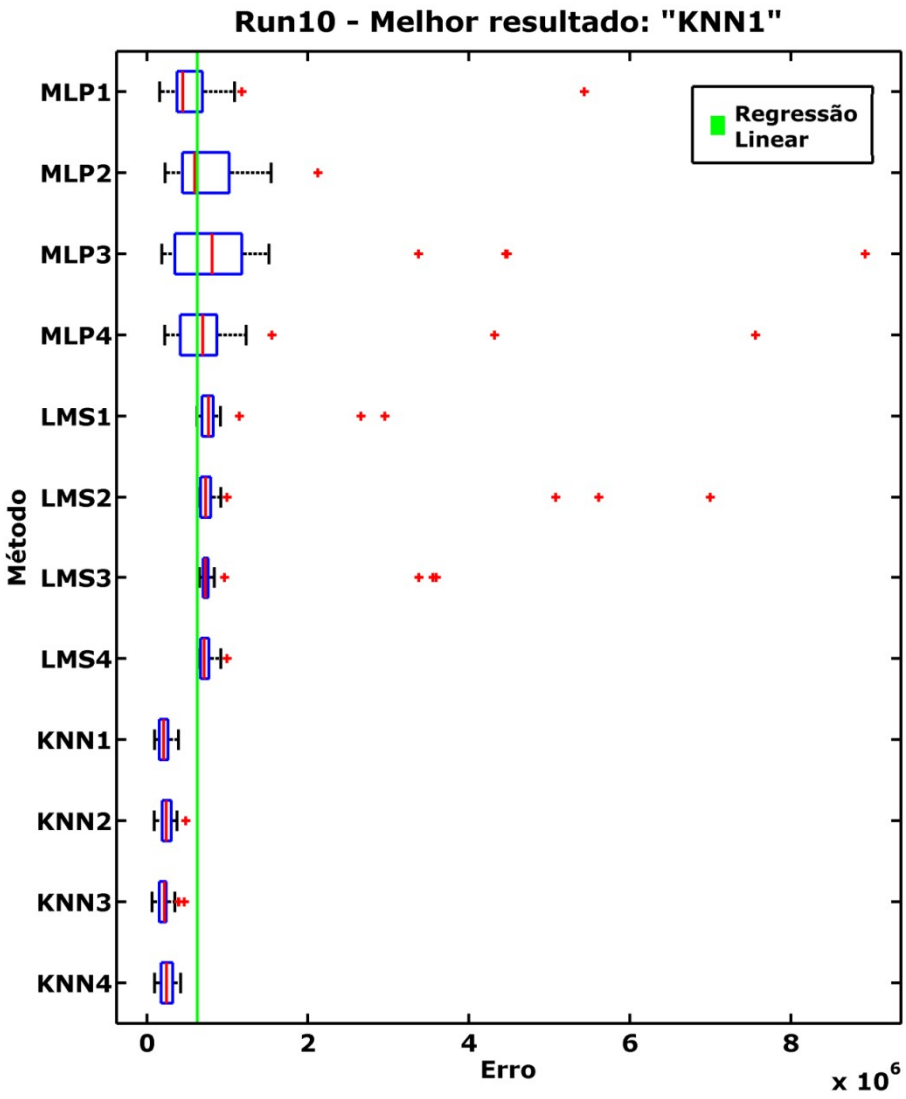

Figura 5-8 – Comparativo dos modelos com dados de Imóveis (apartamento), sem PCA

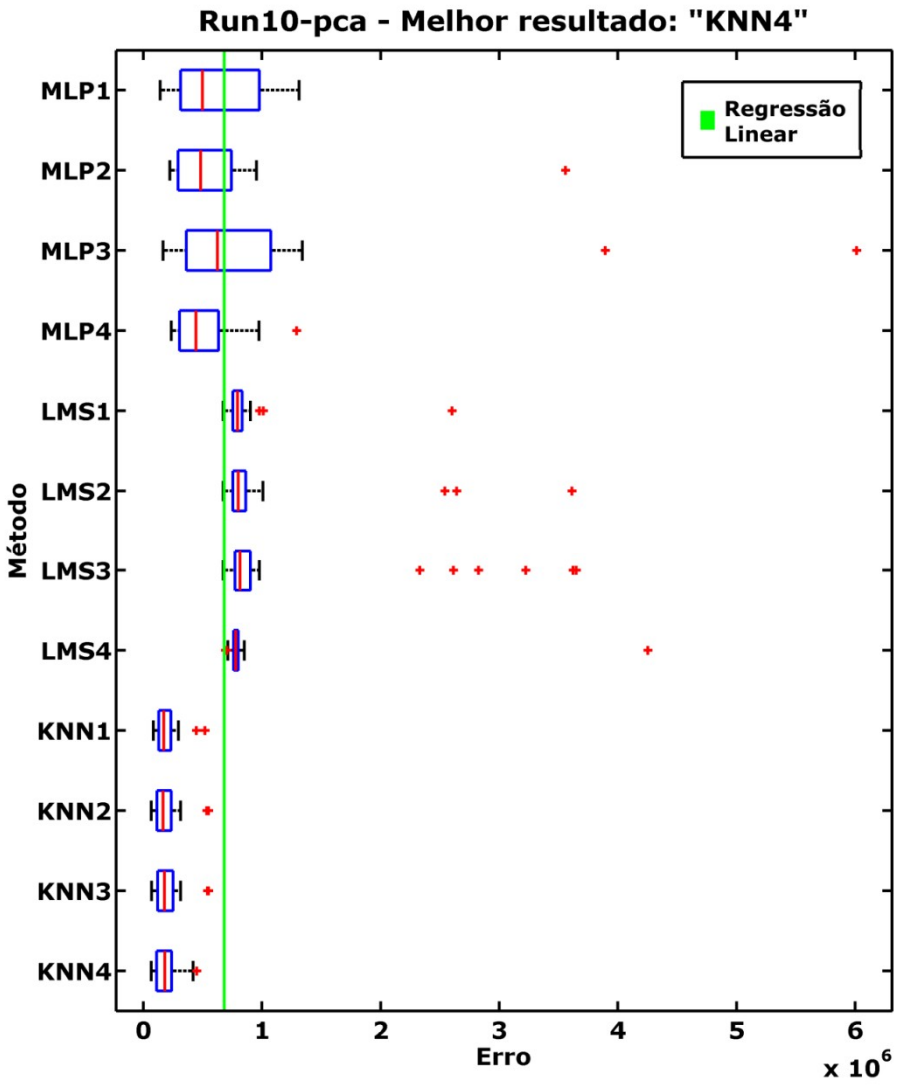

Figura 5-9 – Comparativo dos modelos com dados de Imóveis (apartamento), com PCA

Tabela 5-17 – Dados estatísticos do resultado dos dados de Imóveis (apartamento), sem PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	1.6104e+005	9.3076e+005	4.4642e+005	8 +4 +8 = 20
MLP2	2.2770e+005	4.4148e+005	5.9566e+005	5 +7 +7 = 19
MLP3	1.8805e+005	1.8244e+006	8.1004e+005	7 +1 +1 = 09
MLP4	2.2257e+005	1.4389e+006	6.9583e+005	6 +3 +6 = 15
LMS1	6.2017e+005	5.2994e+005	7.6555e+005	4 +6 +2 = 12
LMS2	6.4607e+005	1.5985e+006	7.2844e+005	2 +2 +3 = 07
LMS3	6.5544e+005	8.5013e+005	7.2544e+005	1 +5 +4 = 10
LMS4	6.4596e+005	8.1545e+004	7.1174e+005	3 +11 +5 = 19
KNN1	**95000**	**6.9332e+004**	**207500**	**10 +12 +12 = 34**
KNN2	93000	8.3910e+004	239500	11 +10 +10 = 31
KNN3	65000	9.3104e+004	218000	12 +8 +11 = 31
KNN4	96000	8.8103e+004	246500	9 +9 +9 = 27

Fonte: Dados calculados

Tabela 5-18 – Dados estatísticos do resultado dos dados de Imóveis (apartamento), com PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	1.4353e+005	3.7268e+005	4.9823e+005	8 +6 +6 = 20
MLP2	2.2294e+005	6.0511e+005	4.8467e+005	6 +5 +7 = 18
MLP3	1.6748e+005	1.1713e+006	6.2452e+005	7 +1 +5 = 13
MLP4	2.3742e+005	2.5735e+005	4.4314e+005	5 +8 +8 = 21
LMS1	6.7277e+005	3.3649e+005	7.9543e+005	2 +7 +3 = 12
LMS2	6.7259e+005	6.7152e+005	8.0105e+005	4 +3 +2 = 09
LMS3	6.7264e+005	9.4224e+005	8.1623e+005	3 +2 +1 = 06
LMS4	6.9048e+005	6.3572e+005	7.7878e+005	1 +4 +4 = 09
KNN1	86000	9.8340e+004	174000	9 +11 +11 = 31
KNN2	67000	1.1718e+005	167500	11 +9 +12 = 32
KNN3	69000	1.1370e+005	179000	10 +10 +10 = 30
KNN4	**67000**	**9.5978e+004**	**182500**	**12 +12 +9 = 33**

Fonte: Dados calculados

Pelo comparativo no gráfico e das tabelas, podemos definir que o KNN4 com o uso do PCA obteve um bom resultado.

A Figura 5-10 mostra a evolução do AG e seu resultado final. A Tabela 5-19,

Tabela 5-20, Tabela 5-21, Tabela 5-22, Tabela 5-23 e a Tabela 5-24 mostram os resultados do AG em todos os modelos.

Tabela 5-19 – Parâmetros encontrados pelo AG - Imóveis (apartamento) sem PCA - MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	28	0.1400	7	"garagem", "área", "peças", "salas", "dormitório", "banheiro", "dist. escola", "dist. hosp.", "dist. merc.", "acab.", "revest. préd", "idade real"
MLP2	53	0.2650	11.4000	"garagem", "local.", "andar", "peças", "salas", "dormitório", "suíte", "banheiro", "dist. escola", "acab.", "conservação", "idade real", "idade aparen"
MLP3	31	0.1550	7.4000	"pos. do apto", "elevador", "área", "andar", "peças", "salas", "dormitório", "suíte", "banheiro", "dep. de emp.", "dist. hosp.", "acab.", "revest. préd", "idade real"
MLP4	61	0.3050	9.4000	"pos. do apto", "garagem", "área", "peças", "dormitório", "suíte", "banheiro", "dep. de emp.", "dist. escola", "dist. hosp.", "conservação", "idade aparen"

Fonte: Dados calculados

Tabela 5-20 – Parâmetros encontrados pelo AG - Imóveis (apartamento) sem PCA - LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.2150	5.6000	"pos. do apto", "garagem", "local.", "área", "andar", "peças", "salas", "dormitório", "suíte", "banheiro", "dist. escola", "dist. hosp.", "idade real"
LMS2	0.1200	8.4000	"pos. do apto", "garagem", "local.", "área", "salas", "dormitório", "suíte", "banheiro", "dist. escola", "acab.", "revest. préd", "idade real"
LMS3	0.2150	2	"pos. do apto", "garagem", "local.", "área", "dormitório", "suíte", "dep. de emp.", "dist. escola", "acab.", "revest. préd", "idade real"
LMS4	0.1650	4	"pos. do apto", "garagem", "local.", "área", "salas", "dormitório", "suíte", "banheiro", "dist. escola", "acab.", "revest. préd", "idade real"

Fonte: Dados calculados

Tabela 5-21 – Parâmetros encontrados pelo AG - Imóveis (apartamento) sem PCA – KNN

Modelo	K	Variáveis
KNN1	7	"área", "pavimento", "peças", "suíte", "banheiro", "dist. hosp.", "revest. préd", "idade real", "idade aparen"
KNN2	2	"área", "pavimento", "peças", "dormitório", "suíte", "banheiro", "dist. merc.", "revest. préd", "conservação", "idade aparen"
KNN3	3	"elevador", "área", "pavimento", "peças", "suíte", "dist. escola", "dist. hosp.", "acab.", "revest. préd", "conservação", "idade real"
KNN4	5	"elevador", "local.", "área", "dormitório", "suíte", "dist. escola", "acab.", "revest. préd", "idade real", "idade aparen"

Fonte: Dados calculados

Tabela 5-22 – Parâmetros encontrados pelo AG - Imóveis (apartamento) com PCA – MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	52	0.2600	12.8000	1,3,4,5,7,8,11,15 - (70.0554%)
MLP2	50	0.2500	9.4000	1,2,4,5,7,8,9,11,12,13,14 - (77.0725%)
MLP3	39	0.1950	3.8000	1,3,4,5,8,11,12,13,14 - (69.2197%)
MLP4	50	0.2500	9.4000	1,3,4,7,9,10,12,14,15 - (67.1289%)

Fonte: Dados calculados

Tabela 5-23 – Parâmetros encontrados pelo AG - Imóveis (apartamento) com PCA – LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.1350	13.4000	1,2,3,4,5,6,7,9,10,11,13,14,15 - (92.8947%)
LMS2	0.2050	13.6000	1,2,3,4,5,6,7,9,10,11,13,14,15 - (92.8947%)
LMS3	0.0750	8.6000	1,2,3,4,5,6,7,9,10,11,13,14,15 - (92.8947%)
LMS4	0.0400	11.2000	1,4,5,6,7,8,9,10,11,12,13,14 - (71.3607%)

Fonte: Dados calculados

Tabela 5-24 – Parâmetros encontrados pelo AG - Imóveis (apartamento) com PCA – KNN

Modelo	K	Variáveis
KNN1	7	1,4,10,11,14,15 - (48.2152%)
KNN2	6	1,3,5,8 - (53.6774%)
KNN3	7	1,3,4,11,12,14,15 - (59.3042%)
KNN4	1	1,3,4,5,9,12,13,15 - (66.3926%)

Fonte: Dados calculados

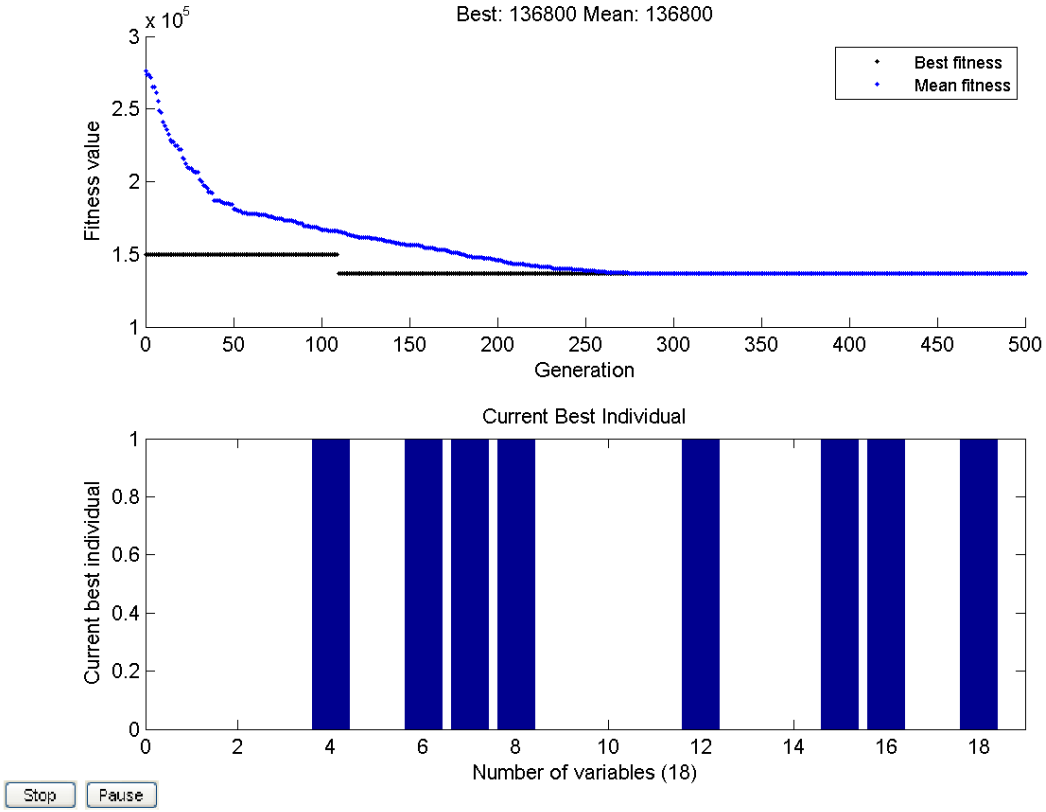

Figura 5-10 – Evolução do AG do KNN4 com PCA e seu resultado final

Dados de Preço das Casas

Pelas Figura 5-11 e Figura 5-12 e avaliando a mediana, o valor mínimo e a dispersão, podemos definir que o LMS02 sem o uso do PCA obteve um bom resultado.

O tempo de execução do processo foi de 02:07:47 e 01:39:12, sem PCA e com PCA respectivamente.

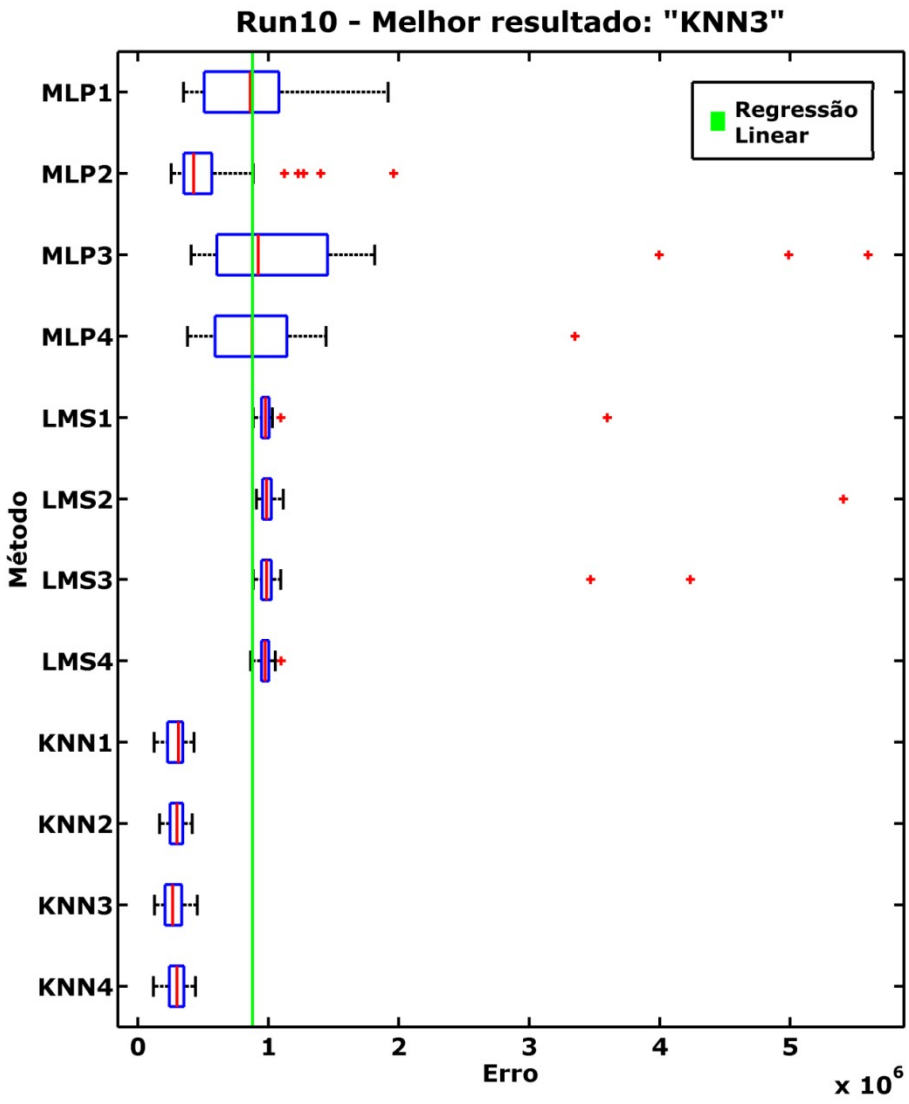

Figura 5-11 – Comparativo dos modelos com dados de Imóveis (casa), sem PCA

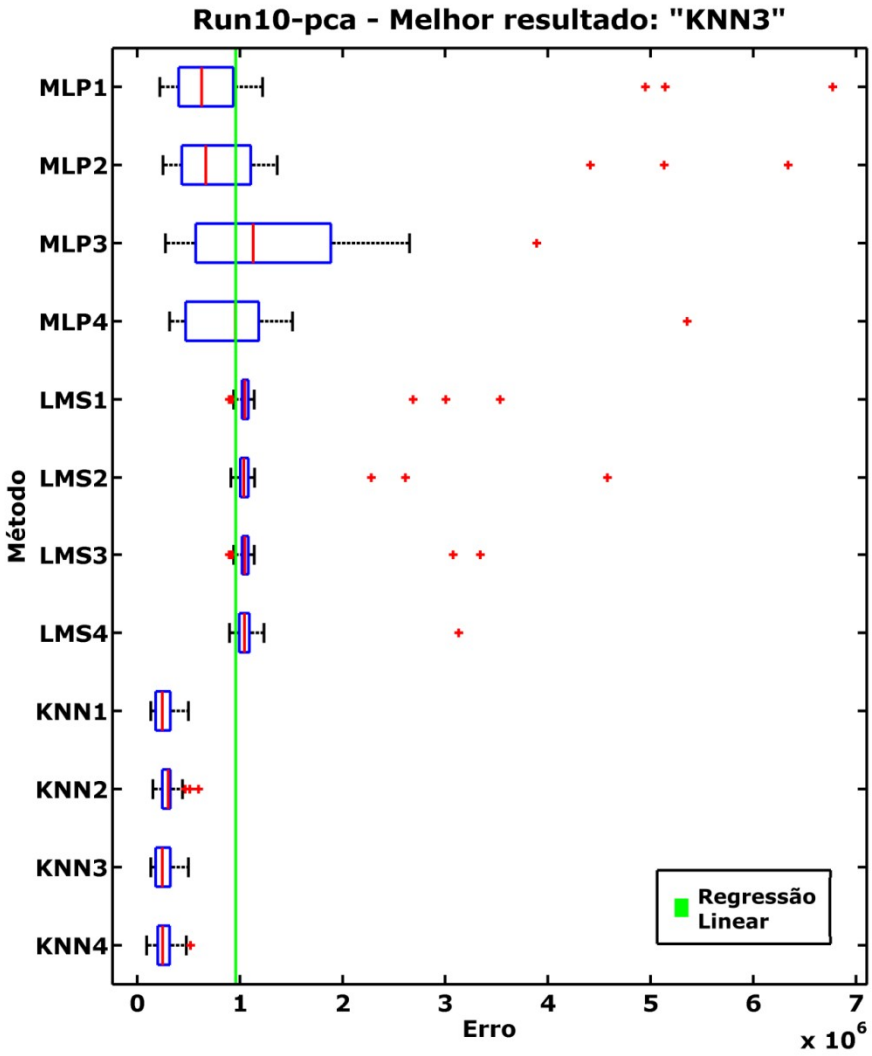

Figura 5-12 – Comparativo dos modelos com dados de Imóveis (casa), com PCA

Tabela 5-25 – Dados estatísticos do resultado dos dados de Imóveis (casa), sem PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	3.5094e+005	4.2153e+005	8.6162e+005	7 +6 +7 = 20
MLP2	2.5671e+005	4.0812e+005	4.2975e+005	8 +7 +8 = 23
MLP3	4.1052e+005	1.2788e+006	9.2389e+005	5 +1 +5 = 11
MLP4	3.8243e+005	5.5560e+005	8.7567e+005	6 +4 +6 = 16
LMS1	8.8442e+005	4.7996e+005	9.7828e+005	2 +5 +3 = 10
LMS2	9.0868e+005	8.0747e+005	9.8626e+005	1 +2 +2 = 05
LMS3	8.8438e+005	7.3632e+005	9.8642e+005	3 +3 +1 = 07
LMS4	8.6571e+005	4.6669e+004	9.7571e+005	4 +12 +4 = 20
KNN1	125000	8.0888e+004	313000	11 +9 +9 = 29
KNN2	167000	6.6646e+004	301000	9 +11 +10 = 30
KNN3	**128000**	**8.0016e+004**	**267500**	**10 +10 +12 = 32**
KNN4	120000	8.1376e+004	301000	12 +8 +11 = 31

Fonte: Dados calculados

Tabela 5-26 – Dados estatísticos do resultado dos dados de Imóveis (casa), com PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	2.2079e+005	1.5758e+006	6.2646e+005	8 +1 +8 = 17
MLP2	2.4976e+005	1.4530e+006	6.6729e+005	7 +2 +7 = 16
MLP3	2.7644e+005	8.6610e+005	1.1314e+006	6 +4 +1 = 11
MLP4	3.1623e+005	9.0834e+005	9.5822e+005	5 +3 +6 = 14
LMS1	8.9722e+005	6.3719e+005	1.0470e+006	4 +6 +3 = 13
LMS2	9.1122e+005	7.2719e+005	1.0398e+006	1 +5 +5 = 11
LMS3	8.9722e+005	5.5687e+005	1.0495e+006	3 +7 +2 = 12
LMS4	8.9991e+005	3.8649e+005	1.0457e+006	2 +8 +4 = 14
KNN1	132000	1.0346e+005	243500	10 +9 +11 = 30
KNN2	152000	1.0171e+005	302000	9 +12 +9 = 30
KNN3	132000	1.0346e+005	243500	11 +10 +12 = 33
KNN4	**92000**	**1.0252e+005**	**248500**	**12 +11 +10 = 33**

Fonte: Dados calculados

Pelo comparativo no gráfico e das tabelas, podemos definir que o KNN4 com o uso do PCA obteve um bom resultado.

A Figura 5-13 mostra a evolução do AG e seu resultado final. A Tabela 5-27, Tabela 5-28,

Tabela 5-29, Tabela 5-30, Tabela 5-31 e a Tabela 5-32 mostram os resultados do AG em todos os modelos.

Tabela 5-27 – Parâmetros encontrados pelo AG - Imóveis (casa) sem PCA - MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	37	0.1850	11.0000	"garagem", "suíte", "banheiro", "dist. mercado", "área const.", "área terreno", "acab.", "cobertura", "estrutura", "conserv.", "dormitório", "lav.", "peças", "idade aparen."
MLP2	42	0.2100	8.8000	"suíte", "edícula", "área const.", "área terreno", "acab.", "estrutura", "conserv.", "piscina", "dormitório", "idade aparen."
MLP3	46	0.2300	5.4000	"bairro", "garagem", "banheiro", "edícula", "dist. mercado", "área const.", "área terreno", "acab.", "cobertura", "estrutura", "conserv.", "dormitório"
MLP4	34	0.1700	1.6000	"suíte", "banheiro", "edícula", "dist. mercado", "área const.", "área terreno", "acab.", "cobertura", "conserv.", "piscina", "dormitório", "lav.", "peças", "idade aparen."

Fonte: Dados calculados

Tabela 5-28 – Parâmetros encontrados pelo AG - Imóveis (casa) sem PCA - LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.0300	5.6000	"bairro", "garagem", "dist. mercado", "área const.", "área terreno", "estrutura", "conserv.", "piscina", "dormitório", "peças"
LMS2	0.0300	8.2000	"bairro", "garagem", "banheiro", "dist. mercado", "área const.", "área terreno", "acab.", "estrutura", "conserv.", "piscina", "dormitório", "peças"
LMS3	0.2800	4.4000	"bairro", "garagem", "dist. mercado", "área const.", "área terreno", "estrutura", "conserv.", "piscina", "dormitório", "peças"
LMS4	0.0500	11.8000	"bairro", "garagem", "dist. mercado", "área const.", "área terreno", "estrutura", "piscina", "dormitório", "peças"

Fonte: Dados calculados

Tabela 5-29 – Parâmetros encontrados pelo AG - Imóveis (casa) sem PCA – KNN

Modelo	K	Variáveis
KNN1	7	"bairro", "garagem", "banheiro", "edícula", "dist. mercado", "área const.", "área terreno", "acab.", "dormitório", "dep.emp."
KNN2	2	"bairro", "suíte", "dist. mercado", "área const.", "área terreno", "acab.", "conserv.", "dormitório", "dep.emp."
KNN3	2	"garagem", "suíte", "área const.", "área terreno", "acab.", "estrutura", "dormitório", "dep.emp.", "peças", "idade aparen."
KNN4	8	"garagem", "suíte", "edícula", "dist. mercado", "área const.", "área terreno", "acab.", "conserv.", "dormitório", "dep.emp.", "peças"

Fonte: Dados calculados

Tabela 5-30 – Parâmetros encontrados pelo AG - Imóveis (casa) com PCA – MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	58	0.2900	1.8000	1,2,3,4,6,7,10,14 - (75.8969%)
MLP2	60	0.3000	9.0000	1,2,3,4,5,7,8,10,12,13,14 - (85.3889%)
MLP3	16	0.0800	9.0000	1,2,4,5,6,7,8,10,11,14 - (79.1528%)
MLP4	40	0.2000	12.2000	1,2,4,7,9,11,14 - (64.7601%)

Fonte: Dados calculados

Tabela 5-31 – Parâmetros encontrados pelo AG - Imóveis (casa) com PCA – LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.2450	3.6000	1,2,4,5,6,7,8,9,11,12,14 - (82.2122%)
LMS2	0.0350	11.2000	1,2,3,5,6,7,8,9,10,11,14 - (85.4177%)
LMS3	0.1450	5.2000	1,2,4,5,6,7,8,9,11,12,14 - (82.2122%)
LMS4	0.1450	12.8000	1,2,5,6,7,8,9,11,13,14 - (74.4795%)

Fonte: Dados calculados

Tabela 5-32 – Parâmetros encontrados pelo AG - Imóveis (casa) com PCA – KNN

Modelo	K	Variáveis
KNN1	2	1,2,3,10,12,13,14 - (63.5554%)
KNN2	7	1,4,6,11,12,14 - (50.0312%)
KNN3	3	1,2,3,10,12,13,14 - (63.5554%)
KNN4	4	1,2,5,7,12,14 - (59.8143%)

Fonte: Dados calculados

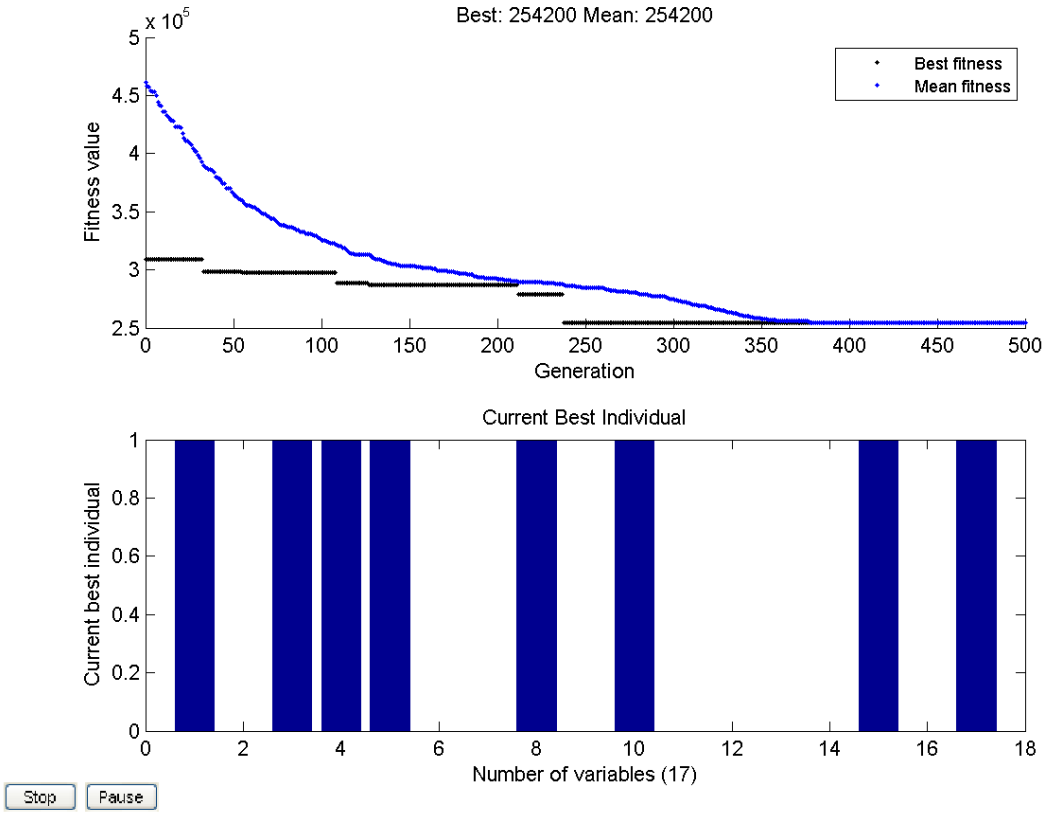

Figura 5-13 – Evolução do AG do KNN3 sem PCA e seu resultado final

Dados de Preços dos Terrenos

Pelas Figura 5-14 e Figura 5-15 e avaliando a mediana, o valor mínimo e a dispersão, podemos definir que o KNN4 sem o uso do PCA obteve um bom resultado. Nesse caso, podemos conferir também que todas as execuções do modelo KNN sem utilização do PCA tiveram o mesmo resultado.

O tempo de execução do processo foi de 01:19:09 e 01:04:39, sem PCA e com PCA respectivamente.

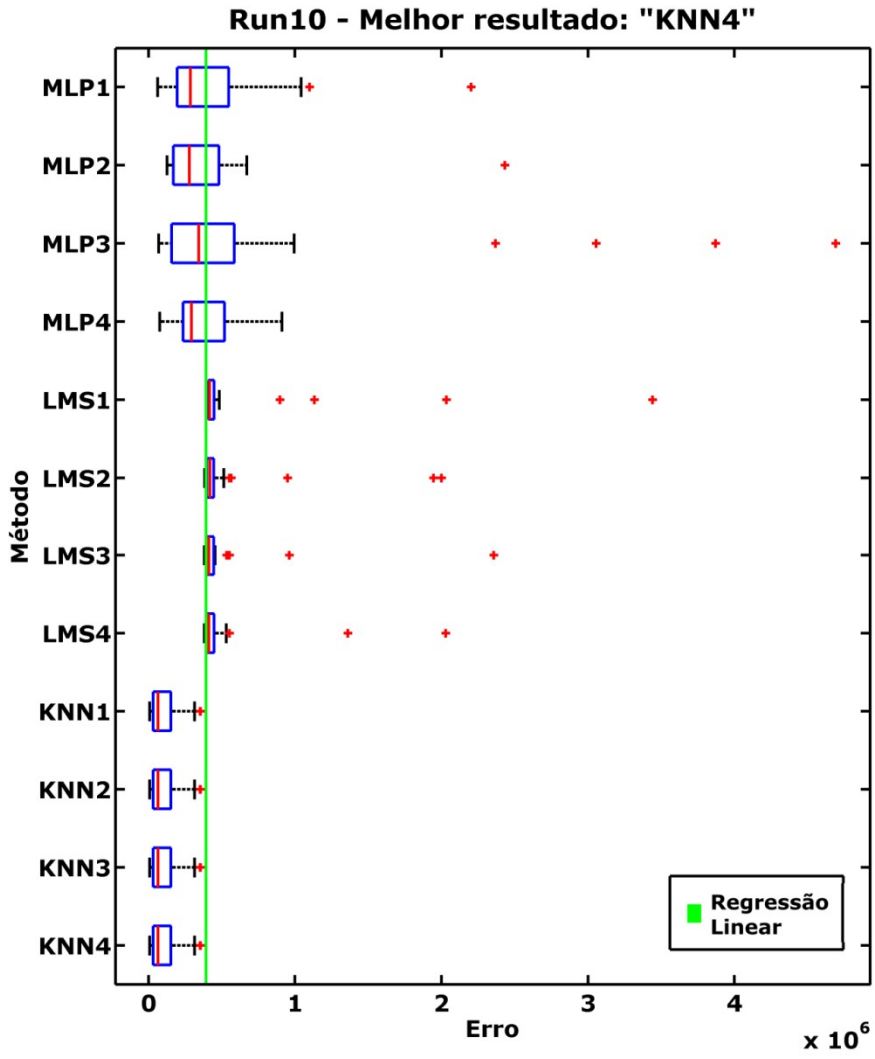

Figura 5-14 – Comparativo dos modelos com dados de Imóveis (terreno), sem PCA

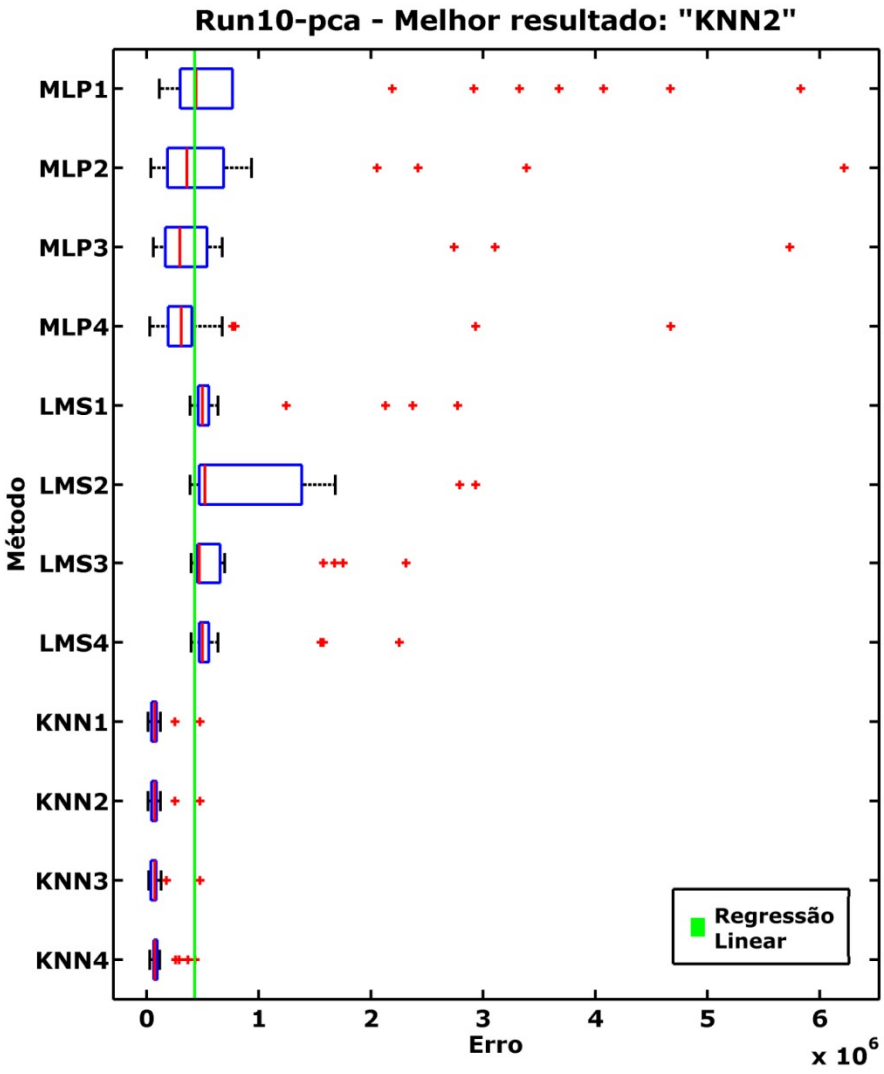

Figura 5-15 – Comparativo dos modelos com dados de Imóveis (terreno), com PCA

Tabela 5-33 – Dados estatísticos do resultado dos dados de Imóveis (terreno), sem PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	6.2475e+004	4.2342e+005	2.8643e+005	8 +3 +7 = 18
MLP2	1.2780e+005	4.2339e+005	2.7833e+005	5 +4 +8 = 17
MLP3	7.0856e+004	1.1587e+006	3.4251e+005	7 +1 +5 = 13
MLP4	7.6960e+004	2.0916e+005	2.9332e+005	6 +8 +6 = 20
LMS1	3.9738e+005	6.2457e+005	4.1593e+005	1 +2 +2 = 05
LMS2	3.8252e+005	4.0152e+005	4.2026e+005	2 +5 +1 = 08
LMS3	3.8075e+005	3.6578e+005	4.1235e+005	4 +6 +3 = 13
LMS4	3.8075e+005	3.3718e+005	4.1174e+005	3 +7 +4 = 14
KNN1	8000	1.0084e+005	64000	9 +9 +9 = 27
KNN2	8000	1.0084e+005	64000	10 +10 +10 = 30
KNN3	8000	1.0084e+005	64000	11 +11 +11 = 33
KNN4	**8000**	**1.0084e+005**	**64000**	**12 +12 +12 = 36**

Fonte: Dados calculados

Tabela 5-34 – Dados estatísticos do resultado dos dados de Imóveis (terreno), com PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	1.1568e+005	1.5811e+006	4.4368e+005	5 +1 +5 = 11
MLP2	4.0613e+004	1.2687e+006	3.6155e+005	7 +2 +6 = 15
MLP3	6.1230e+004	1.1790e+006	2.9771e+005	6 +3 +8 = 17
MLP4	2.9652e+004	9.3244e+005	3.1074e+005	9 +4 +7 = 20
LMS1	3.8992e+005	6.0550e+005	5.0054e+005	3 +6 +2 = 11
LMS2	3.8992e+005	6.9611e+005	5.2193e+005	4 +5 +1 = 10
LMS3	3.9671e+005	4.7904e+005	4.7148e+005	2 +7 +4 = 13
LMS4	3.9843e+005	4.1293e+005	5.0053e+005	1 +8 +3 = 12
KNN1	14000	8.5606e+004	73500	11 +10 +11 = 32
KNN2	**14000**	**8.5606e+004**	**73500**	**12 +11 +12 = 35**
KNN3	19000	8.2568e+004	74000	10 +12 +9 = 31
KNN4	30000	9.8592e+004	74000	8 +9 +10 = 27

Fonte: Dados calculados

Pelo comparativo no gráfico e das tabelas, podemos definir que o KNN4 sem o uso do PCA obteve um bom resultado.

A Figura 5-16 mostra a evolução do AG e seu resultado final. A Tabela 5-35, Tabela 5-36, Tabela 5-37,

Tabela 5-38, Tabela 5-39, Tabela 5-40 mostram os resultados do AG em todos os modelos.

Tabela 5-35 – Parâmetros encontrados pelo AG - Imóveis (terreno) sem PCA - MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	59	0.2950	12.2000	"setor comer.", "pólo", "frente", "área do ter.", "proteção", "pavimentação"
MLP2	32	0.1600	3.8000	"setor comer.", "pólo", "proteção", "inclinado", "pavimentação"
MLP3	35	0.1750	12.8000	"localização", "setor comer.", "pólo", "proteção", "pavimentação"
MLP4	37	0.1850	2.8000	"localização", "setor comer.", "pólo", "proteção", "inclinado", "pavimentação"

Fonte: Dados calculados

Tabela 5-36 – Parâmetros encontrados pelo AG - Imóveis (terreno) sem PCA - LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.0050	11.4000	"localização", "setor comer.", "pavimentação"
LMS2	0.0250	12.4000	"localização", "setor comer.", "pólo", "proteção", "pavimentação"
LMS3	0.2250	4.6000	"localização", "setor comer.", "proteção", "pavimentação"
LMS4	0.2200	6.8000	"localização", "setor comer.", "proteção", "pavimentação"

Fonte: Dados calculados

Tabela 5-37 – Parâmetros encontrados pelo AG - Imóveis (terreno) sem PCA – KNN

Modelo	K	Variáveis
KNN1	1	"localização", "setor comer.", "pólo", "área do ter.", "inclinado", "posição"
KNN2	8	"localização", "setor comer.", "pólo", "área do ter.", "inclinado", "posição"
KNN3	4	"localização", "setor comer.", "pólo", "área do ter.", "inclinado", "posição"
KNN4	4	"localização", "setor comer.", "pólo", "área do ter.", "inclinado", "posição"

Fonte: Dados calculados

Tabela 5-38 – Parâmetros encontrados pelo AG - Imóveis (terreno) com PCA – MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	52	0.2600	1.6000	2,4,5,7 - (46.6205%)
MLP2	26	0.1300	2.4000	1,2,3,4,5,7,8 - (91.6441%)
MLP3	32	0.1600	12.0000	1,2,3,4,6,7 - (87.5063%)
MLP4	37	0.1850	1.8000	1,2,4,5 - (73.6176%)

Fonte: Dados calculados

Tabela 5-39 – Parâmetros encontrados pelo AG - Imóveis (terreno) com PCA – LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.0550	3	1,2,3,4,6,7,8 - (90.1700%)
LMS2	0.2500	12.2000	1,2,3,4,6,7,8 - (90.1700%)
LMS3	0.1350	9.4000	1,2,3,4,5,6,7,8 - (97.6252%)
LMS4	0.2100	9.0000	1,2,3,4,6,7,8 - (90.1700%)

Fonte: Dados calculados

Tabela 5-40 – Parâmetros encontrados pelo AG - Imóveis (terreno) com PCA – KNN

Modelo	K	Variáveis
KNN1	1	1,2,5,6,7 - (72.5965%)
KNN2	2	1,2,5,6,7 - (72.5965%)
KNN3	4	1,2,3,5,7,8 - (81.5813%)
KNN4	5	6,7 - (09.0417%)

Fonte: Dados calculados

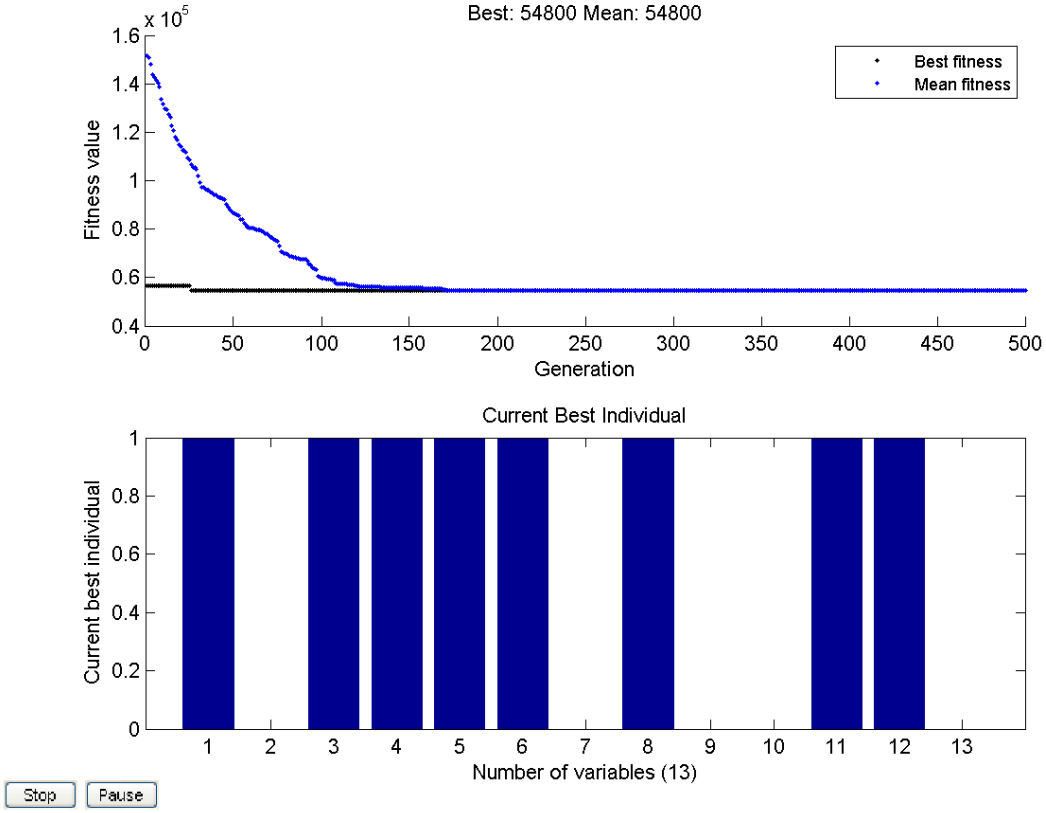

Figura 5-16 – Evolução do AG do KNN4 sem PCA e seu resultado final

Avaliação dos Resultados

O resultado do banco de dados de preço de imóveis foi curioso talvez devido ao pequeno número de indivíduos nas amostras.

Podemos ver que o KNN demonstrou um resultado bem superior aos demais modelos.

A conseqüência do uso do PCA foi diferente em cada um dos conjuntos de dados. No banco de terrenos e no banco de casas ele não ajudou, porém no banco de dados de apartamentos ele ajudou, melhorando o valor mínimo.

Comparação com outros trabalhos

Comparando nossos melhores resultados (KNN4 com PCA, KNN3 sem PCA, KNN4 sem PCA) com o resultado do trabalho apresentado em [41].

Apesar de o modelo ser um RNA não foi utilizado grupo para validação cruzada. Os grupos foram divididos 3/4 para treinamento e 1/4 para teste. Como nossos modelos vencedores foram KNN, não teremos necessidade de grupo de validação cruzada, logo poderemos utilizar a mesma divisão de grupos.

Primeiro iremos avaliar os dados dos apartamentos:

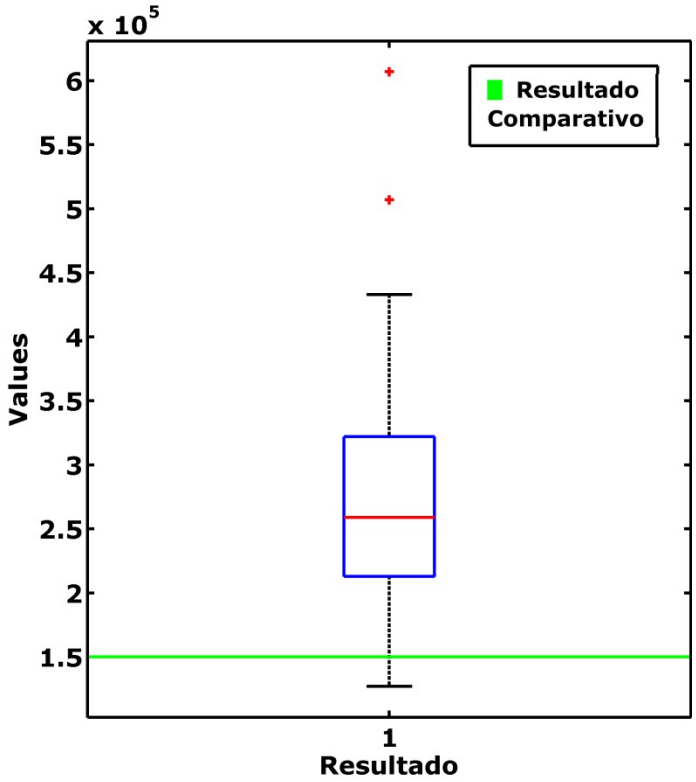

Figura 5-17 – Comparação dos dados de imóveis - Apartamento

Como podemos ver pela Figura 5-17, tivemos um resultado melhor apenas com determinadas amostras. Como no trabalho [41] não foi publicado resultados para conjuntos aleatórios para os agrupamentos, publicou apenas o seu melhor resultado, podemos dizer que tivemos casos melhores com o uso do KNN.

Agora vamos verificar o resultado para os dados das casas:

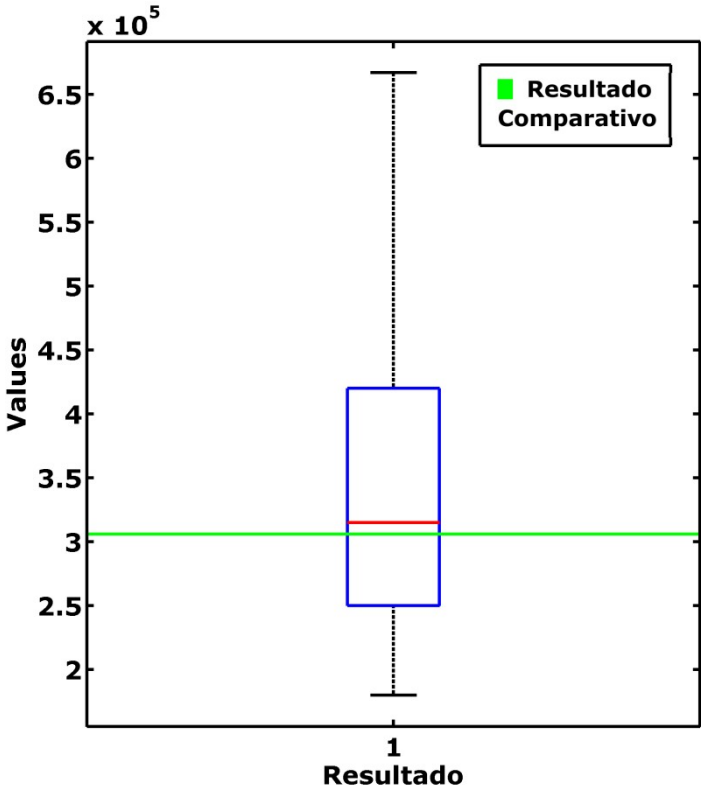

Figura 5-18 – Comparação dos dados de imóveis - Casas

Como podemos ver pela Figura 5-18, tivemos um resultado ainda melhor que no anterior dos apartamentos, a mediana ficou próximo do resultado do trabalho [41], ou seja em quase metade dos agrupamentos tivemos um resultado melhor com o uso do KNN.

E agora por último os dados dos Terrenos:

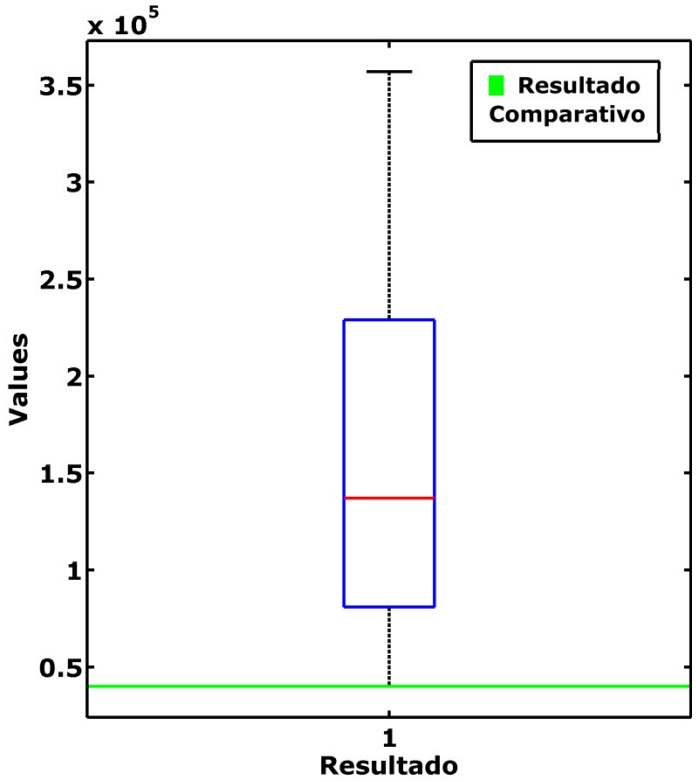

Figura 5-19 – Comparação dos dados de imóveis - Terrenos

Como podemos ver pela Figura 5-19, tivemos como melhor resultado apenas comparado ao melhor do trabalho [41]. Ou seja, apenas um agrupamento dos dados foi igual ao resultado obtido no [41], os demais foram piores.

5.4. Dados ABALONE

A função de erro utilizada no cálculo relativo à base de dados ABALONE foi o MSE, representada da equação 5-1.

O tempo de execução do processo foi de 02:58:04 e 02:10:36, sem PCA e com PCA respectivamente.

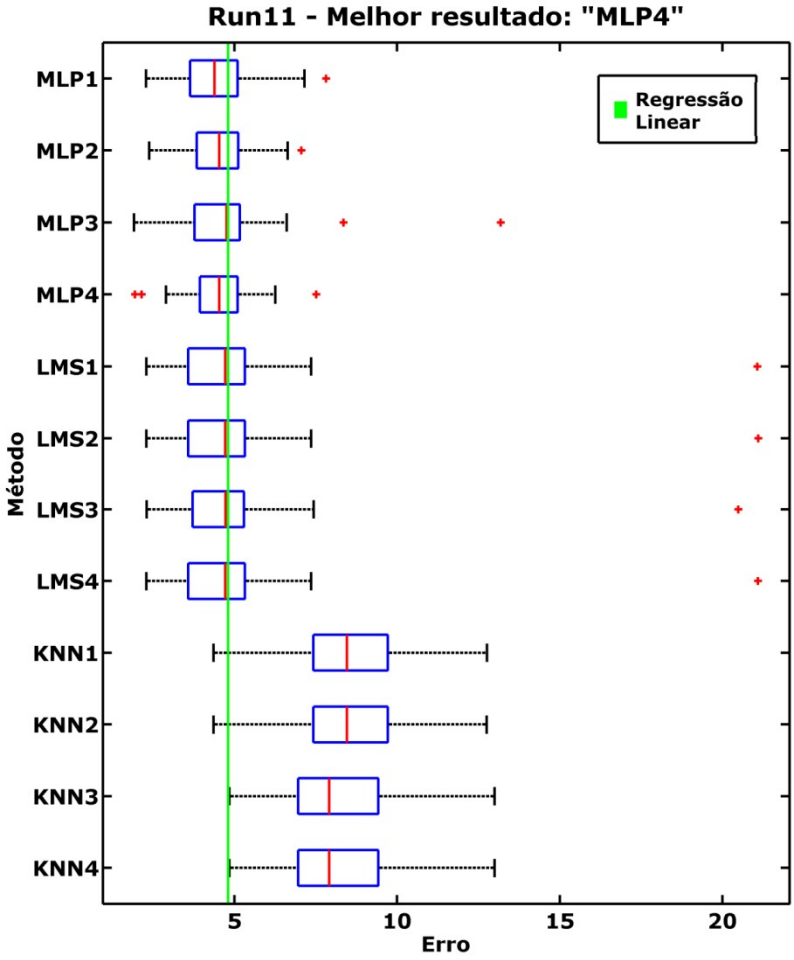

Figura 5-20 – Comparativo dos modelos da base ABALONE, sem PCA

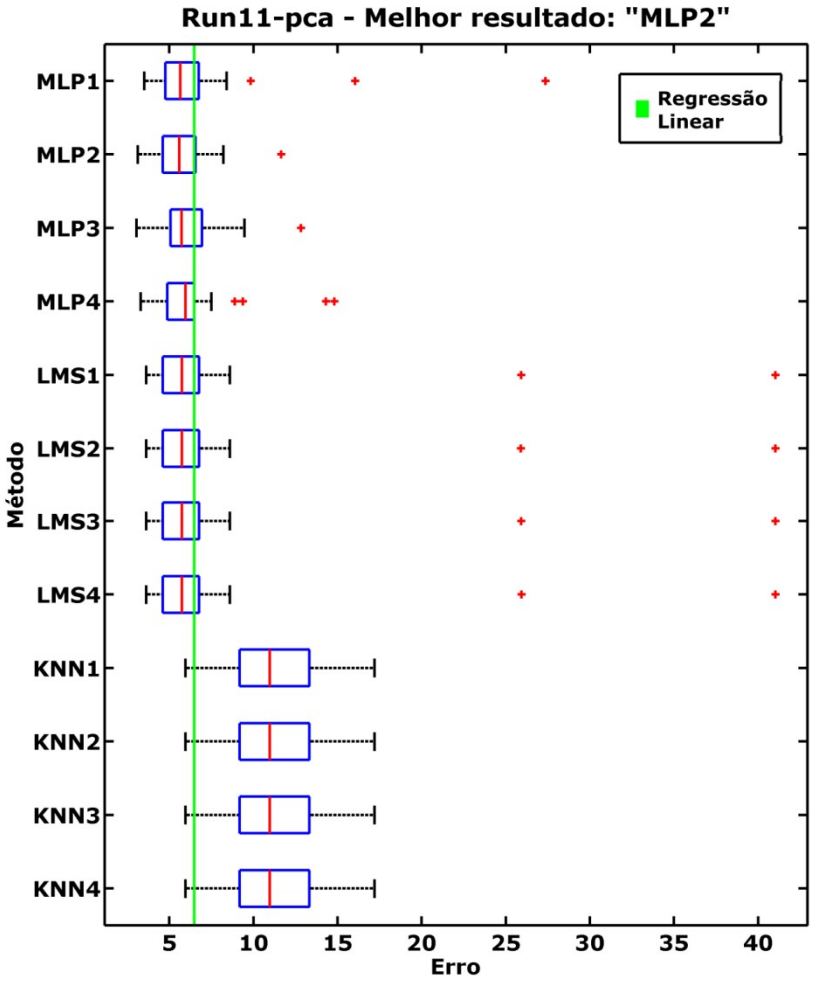

Figura 5-21 – Comparativo dos modelos da base ABALONE, com PCA

Tabela 5-41 – Dados estatísticos do resultado dos dados ABALONE, sem PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	2.2803	1.2288	4.3841	10 +10 +12 = 32
MLP2	2.3817	1.1648	4.5330	5 +11 +11 = 27
MLP3	1.9134	2.0038	4.7552	12 +9 +5 = 26
MLP4	**1.9467**	**1.1428**	**4.5364**	**11 +12 +10 = 33**
LMS1	2.2934	3.2208	4.7232	8 +3 +9 = 20
LMS2	2.2935	3.2252	4.7233	7 +1 +8 = 16
LMS3	2.3047	3.1253	4.7247	6 +4 +6 = 16
LMS4	2.2932	3.2233	4.7233	9 +2 +7 = 18
KNN1	4.3600	2.1474	8.4550	3 +5 +1 = 09
KNN2	4.3600	2.1440	8.4550	4 +6 +2 = 12
KNN3	4.8500	2.0080	7.9100	1 +7 +3 = 11
KNN4	4.8500	2.0080	7.9100	2 +8 +4 = 14

Fonte: Dados calculados

Tabela 5-42 – Dados estatísticos do resultado dos dados ABALONE, com PCA

Modelo	Valor Mínimo	Desv. Padrão	Mediana	Pontuação
MLP1	3.5165	4.5299	5.6433	9 +5 +11 = 25
MLP2	**3.1223**	**1.7108**	**5.5829**	**11 +12 +12 = 35**
MLP3	3.0441	1.8730	5.7238	12 +11 +10 = 33
MLP4	3.2905	2.6059	5.9628	10 +10 +5 = 25
LMS1	3.6261	7.4221	5.7517	5 +2 +7 = 14
LMS2	3.6261	7.4219	5.7517	6 +3 +8 = 17
LMS3	3.6259	7.4214	5.7517	8 +4 +9 = 21
LMS4	3.6261	7.4224	5.7517	7 +1 +6 = 14
KNN1	5.9500	2.7675	10.9750	1 +6 +1 = 08
KNN2	5.9500	2.7675	10.9750	2 +7 +2 = 11
KNN3	5.9500	2.7675	10.9750	3 +8 +3 = 14
KNN4	5.9500	2.7675	10.9750	4 +9 +4 = 17

Fonte: Dados calculados

Pelo comparativo no gráfico e das tabelas, podemos definir que o MLP4 sem o uso do PCA obteve um bom resultado.

A Figura 5-22 mostra a evolução do AG e seu resultado final. A Tabela 5-43, Tabela 5-44,

Tabela 5-45, Tabela 5-46, Tabela 5-47 e a Tabela 5-48 mostram os resultados do AG em todos os modelos.

Tabela 5-43 – Parâmetros encontrados pelo AG - ABALONE sem PCA - MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	7	0.0350	3.6000	"Infant", "Length", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
MLP2	7	0.0350	7.4000	"Infant", "Height", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
MLP3	6	0.0300	2	"Height", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
MLP4	3	0.0150	13.6000	"Infant", "Length", "Whole_Weight", "Shucked_Weight", "Shell_Weight"

Fonte: Dados calculados

Tabela 5-44 – Parâmetros encontrados pelo AG - ABALONE sem PCA - LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.1250	2	"Infant", "Length", "Height", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
LMS2	0.0200	1.4000	"Infant", "Length", "Height", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
LMS3	0.0800	12.8000	"Infant", "Diameter", "Height", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
LMS4	0.2500	8.2000	"Infant", "Length", "Height", "Whole_Weight", "Shucked_Weight", "Shell_Weight"

Fonte: Dados calculados

Tabela 5-45 – Parâmetros encontrados pelo AG - ABALONE sem PCA – KNN

Modelo	K	Variáveis
KNN1	4	"Male", "Female", "Infant", "Diameter", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
KNN2	7	"Male", "Infant", "Diameter", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
KNN3	7	"Female", "Diameter", "Whole_Weight", "Shucked_Weight", "Shell_Weight"
KNN4	1	"Female", "Diameter", "Whole_Weight", "Shucked_Weight", "Shell_Weight"

Fonte: Dados calculados

Tabela 5-46 – Parâmetros encontrados pelo AG - ABALONE com PCA – MLP

Modelo	PEs	Mu_dec	Mu_inc	Variáveis
MLP1	17	0.0850	9.8000	2,4 - (17.9722%)
MLP2	29	0.1450	13.4000	2,4 - (17.9722%)
MLP3	22	0.1100	7	2,4 - (17.9722%)
MLP4	20	0.1000	4.8000	2,4 - (17.9722%)

Fonte: Dados calculados

Tabela 5-47 – Parâmetros encontrados pelo AG - ABALONE com PCA – LMS

Modelo	Mu_dec	Mu_inc	Variáveis
LMS1	0.1150	3.6000	1,3,4 - (81.1631%)
LMS2	0.1400	11.2000	1,3,4 - (81.1631%)
LMS3	0.2550	1.6000	1,3,4 - (81.1631%)
LMS4	0.3200	13.6000	1,3,4 - (81.1631%)

Fonte: Dados calculados

Tabela 5-48 – Parâmetros encontrados pelo AG - ABALONE com PCA – KNN

Modelo	K	Variáveis
KNN1	2	2,4 - (17.9722%)
KNN2	4	2,4 - (17.9722%)
KNN3	2	2,4 - (17.9722%)
KNN4	2	2,4 - (17.9722%)

Fonte: Dados calculados

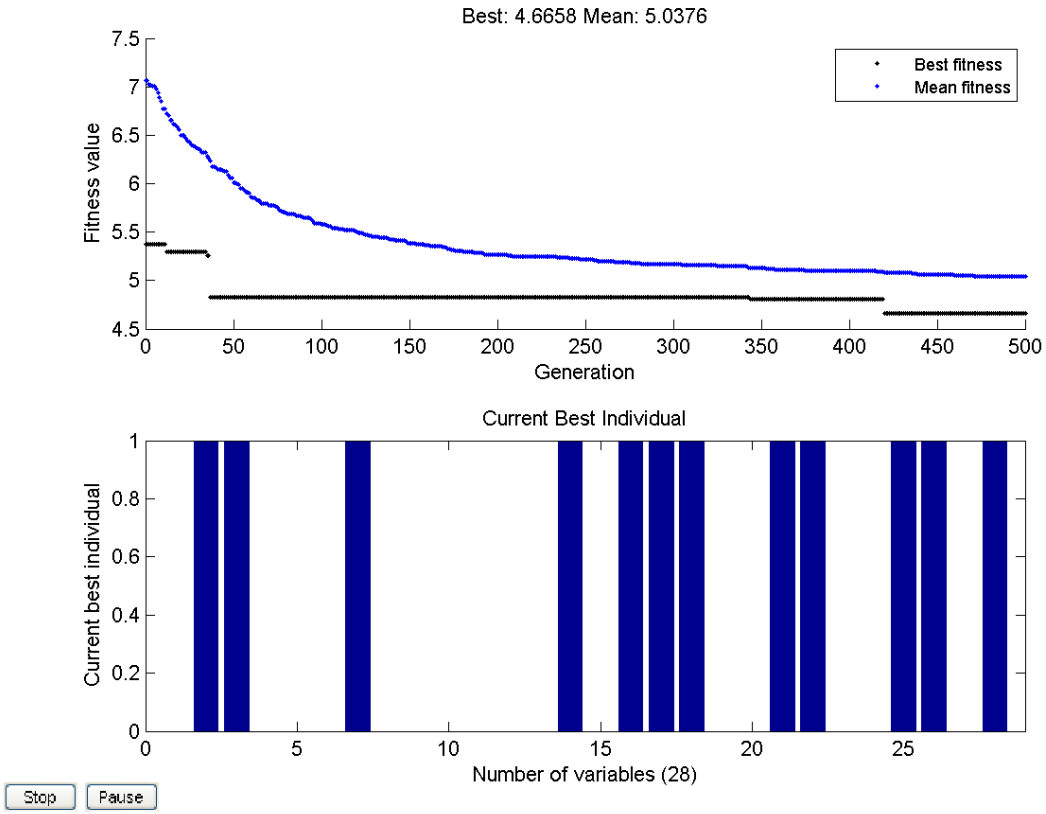

Figura 5-22 – Evolução do AG do MLP4 sem PCA e seu resultado final

Avaliação do Resultado

O banco de dados ABALONE teve um resultado mais simples, onde as RNA disputaram o melhor resultado, como pode ser visto nas Figura 5-20 e Figura 5-21. É uma escolha difícil, porém o modelo do MLP4 sem PCA se mostrou com melhores resultados.

A utilização do PCA acarretou em uma pequena melhora na mediana de todos os modelos, porém piorou o menor valor de erro.

6. Considerações Finais

Considerando que o principal objetivo do trabalho é demonstrar o funcionamento dos diversos modelos de regressão apresentados, nosso trabalho conseguiu um bom resultado, pois mostra como o conjunto de dados influencia em cada modelo e que isso deve ser considerado num estudo mais aprofundado do problema.

As diversas etapas do trabalho mostram as opções que iremos encontrar na utilização de cada um dos modelos e no uso em conjunto do processo de otimização dos parâmetros e do processo de seleção de variáveis para facilitar a definição de cada opção.

Sobre o banco de dados de complexos proteína-ligante, primeiro banco de dados do trabalho, podemos dizer que ainda existe muito a estudar. Pois o resultado mostrou que boa parte das variáveis não está ajudando no ajuste.

Essa informação possivelmente irá direcionar os estudos da simulação para entender onde as demais informações estão influenciando. Esperamos que a ferramenta desenvolvida, bem como o aumento do banco de dados ajude a trazer melhores resultados para a predição da constante de afinidade.

6.1. Conclusões

Algumas observações e conclusões podem ser feitas.

O uso do processo de otimização dos parâmetros foi de grande importância, pois facilita o ajuste dos parâmetros. Como também o uso do processo de seleção de variáveis se mostrou muito interessante, seu resultado nos leva para estudo mais conclusivo da seleção dos atributos dos dados de entreda.

Porém vimos que nem sempre encontramos os melhores resultados. É possível que um trabalho mais apurado no AG torne os resultados melhores.

O uso do PCA nem sempre ajuda no ajuste dos dados. Por isso, fazer testes com e sem o PCA é uma boa opção. Vimos porém que a utilização do mesmo facilita no processo computacional, tornando o processo mais rápido.

Cada um dos modelos se adaptou melhor a cada banco de dados. Até mesmo o simples KNN demonstrou bons resultados. Com isso devemos considerar a ferramenta como uma ajuda na análise de regressão e não como resultado final. Ele irá mostrar qual o próximo caminho a ser seguido pelo pesquisador, apurando assim o seu regressor.

6.2. Sugestões para trabalhos futuros

Uma primeira sugestão seria aumentar o número de modelos disponíveis na ferramenta para regressão de dados tornando a comparação dos modelos mais efetiva e facilitando ainda mais o trabalho do pesquisador.

Para esses modelos pode-se sugerir o uso de Função de Base Radial (RBF) e o uso do algoritmo de máquina de vetores suporte (SVM) para regressão. A literatura mostra uma boa generalização do uso dos dois modelos com um conjunto de dados limitado.

Outra idéia é utilizar algum modelo de clusterização para criarmos grupos mais homogêneos antes do treinamento dos modelos como, por exemplo, o uso da Rede de Kohonen. Essa homogenização torna o processo de treinamento mais simples. Porém, sua utilização deve ser cuidadosa, pois isso irá diminuir o conjunto de elementos em cada grupo de treinamento.

Finalmente, implementar os modelos em uma linguagem de programação livre, possibilitando sua utilização sem o uso do MatLab. Assim como também utilizar recursos de paralelismo, para melhorar o tempo de processamento necessário nos treinamentos de cada modelo.

Referencias

1. **RÊGO, T. G.** *Construção de Funções Empíricas Utilizando Rede Neural para Determinação de Constantes de Afinidade Receptor-Ligante.* Programa de Pós Graduação em Modelagem Computacional, Laboratório Nacional de Computação Científica. Petrópolis. Dissertação de Mestrado. 2008
2. **BRAÚLIO, S. N.** *Proposta de uma metodologia para a avaliação de imóveis urbanos.* UFPR. Curitiba. Dissertação de Mestrado. 2005
3. **ALVES, V.** *Avaliação de imóveis urbanos baseada em métodos estatísticos.* UFPR. Curitiba. Dissertação de Mestrado. 2005
4. **DARWIN, C.** *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life.* 1859.
5. **FOGEL, L. J., OWENS, A. J. e WALSH, M. J.** *Artificial Intelligence through Simulated Evolution.* New York : John Wiley, 1966.
6. **RECHENBERG, I.** *Cybernetic Solution Path of an Experimental Problem.* Farnborough, Hants, 1965.
7. **HOLLAND, J. H.** *Genetic algorithms and the optimal allocation of trials.* SIAM J. Comput., Vol. 2, pp. 88-105. 1973
8. **HOLLAND, J. H.** *Adaptation in Natural and Artificial Systems.* Ann Arbor : University of Michigan Press, 1975.
9. **HOLLAND, J. H.** *Adaptation in Natural and Artificial Systems.* Ann Arbor : University of Michigan Press, 1992.
10. **LEMONGE, A. C. C.** *Aplicação de Algoritmos Genéticos em Otimização Estrutural. Prog. de Engenharia Civil*, COPPE/UFRJ. Tese de Doutorado. 1999.
11. **BERNARDINO, H. S., BARBOSA, H. J. C. e LEMONGE, A. C. C.** *A hybrid genetic algorithm for constrained optimization problems in mechanical engineering*. Proceedings of the 2007 IEEE Congress on Evolutionary Computation. Singapore : IEEE Press, 2007.
12. **LEMONGE, A. C. C. e BARBOSA, H. J. C.** *An adaptive penalty scheme for genetic algorithms in structural optimization*. Int. Journal for Numerical Methods in Engineering, Vol. 59, pp. 703-736. 2004.
13. **SINGH, G. e DEB, K.** *Comparison of multi-modal optimization algorithms based on evolutionary algorithms.* Seattle, WA, USA : GECCO-2006. Genetic And Evolutionary Computation Conference. 2006
14. **LIANG, J. J.** *Problem Definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization.* 2006, pp. 1-24. Online: <http://www.ntu.edu.sg/home/EPNSugan/index_files/CEC06/cec2006.zip>.
15. **QIAN, Y.** *Image interpretation with fuzzy-graph based genetic algorithm.* International Conference on Image Processing. Vol. 1, pp. 545-549. 1999.
16. **NASSIF, N., KAJL, S. e SABOURIN, R.** *Optimization of HVAC control system strategy using two-objective genetic algorithm.* International Journal of HVAC&R Research, Vol. 11. 2005.
17. **BORGES, F. P. e S.** *Otimização via Algoritmo Genético do Processo Construtivo de Estruturas de Concreto Submetidos à Retração Restringida Tendo em.* Universidade Federal do Rio de Janeiro. Rio de Janeiro. 2002.
18. **LINDEN, R.** *Algoritmos Genéticos: Uma Importante Ferramenta da Inteligência Computacional*. Brasport, 2006.
19. **BLICKLE, T. e THIELE, L.** *A Comparison of Selection Schemes Used in Genetic Algorithms.* Zurich : Gloriastrasse 35, 8092, 1995.

20. **VAPNIK, V. N. e CHERVONENKIS, A. Y.** *On the uniform convergence of relative.* Theory of probability and its applications, Vol. 16, pp. 264-280. 1971.
21. **PRINCIPE, J. C., EULIANO, N. R. e LEFEBVRE, W. C.** *Neural And Adaptative Systems.* John Wiley & Sons Inc., 2000.
22. **WIDROW, B. e HOFF JR.** *Adaptative Switching Circuits.* M. E. In Ire Western Electric Show And Convention Record, Vol. 4, pp. 96-104. 1960.
23. **ROSENBLATT, F.** *The Perceptron: A probabilistic model for information storage and organization in the brain.* Psychological Review, Vol. 65, pp. 386-408. 1958.
24. **MINSKY, M.L. e PAPERT, S. A.** *Perceptrons.* Cambridge MA : MIT Press, 1969.
25. **KOLMOGOROV, A. N.** *On the representation of continuous functions of several variables by superposition of continuous functions of one variable and addition.* URSS. Doklady Akademiia Nauk, Vol. 114, pp. 953-956. 1957.
26. **FUNAHASHI, K.** *On the approximate realization of continuous mappings by neural networks.* Neural Networks, Vol. 2, pp. 183-192. 1989.
27. **CYBENKO, G.** *Approximation by superpositions of a sigmoidal function.* Mathematics of Control. Signals and Systems, Vol. 2, pp. 304-314. 1989.
28. **WERBOS, P.** *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Harvard. Ph.D. dissertation. 1974.
29. **RUMELHARD, D., HILTON, G., WILLIAMS, R.** *Learning representations by backpropagation errors.* Nature, Vol. 323, pp. 533-566. 1986.
30. **LEVENBERG, K.** *A Method for the Solution of Certain Non-Linear Problems in Least Squares.* he Quarterly of Applied Mathematics, Vol. 2, pp. 164–168. 1944.
31. **MARQUARDT, D.** *An Algorithm for Least-Squares Estimation of Nonlinear Parameters.* SIAM Journal on Applied Mathematics, Vol. 11, pp. 431–441. 1963.
32. *Gauss–Newton algorithm.* Wikipedia. [Online] [Cited: julho 10, 2008.] http://en.wikipedia.org/wiki/Gauss-Newton_method.
33. *Newton's method.* Wikipedia. [Online] [Cited: julho 10, 2008.] http://en.wikipedia.org/wiki/Newton's_method.
34. **COVER, T. M. e HART, P. E.** *Nearest neighbor pattern classification.* IEEE Trans. Inform. Theory, Vols. IT-13, pp. 21–27. 1968.
35. **SHEPARD, D.** *A two-dimensional interpolation function for irregularly-spaced data*. ACM, Proceedings of the 1968 ACM National Conference, pp. 517-524. 1968.
36. **JACKSON, J. E.** *A User's Guide to Principal.* John Wiley and Sons, p. 592. 1991.
37. **JOLLIFFE, I. T.** *Principal Component Analysis.* Springer, 2002.
38. **KRZANOWSKI, W. J.** *Principles of Multivariate.* Oxford University Press, 1988.
39. **SABER, G. A. F.** *Multivariate Observations.* Wiley, 1984.
40. **DEFILIPPO, S. B. e HIPPERT, H. S.** *Modelagem Da Demanda Residencial De Energia Elétrica Por Meio De Redes Neurais E Algoritmos Genéticos.* Anais do VIII Congresso Brasileiro De Redes Neurais. out 09 a 11, 2007.
41. **MOTA, J. F.** *Um estudo de caso para a determinação do preço de venda de imóveis urbanos via redes neurais artificiais e métodos estatísticos multivariados.* Setor de Ciências Exatas e de Tecnologia, Universidade Federal do Paraná. Curitiba. Dissertação de Mestrado. 2007.

Apêndice A – Bancos de dados utilizados

Dados de Complexos Proteína-Ligante

B	C	D	E	F	G	Saida
95,745	-11,7466	-35,3714	5	2	2	6,74
98,112	-9,02132	-28,8863	3	3	3	3,72
90,43	-10,5621	-26,7606	3	3	1	6,04
81,898	-26,718	-26,3242	8	4	4	4,85
96,09	-66,9633	-20,9483	5	5	3	6,35
85,098	-4,8779	-40,2326	6	3	2	9,43
91,906	-16,3357	-17,2044	6	5	3	6,66
77,025	-2,59077	-37,791	3	1	1	9
83,4	-2,42038	-39,6174	2	0	0	7,68
81,414	-10,6225	-45,1232	6	9	9	7,4
43,492	-31,4298	-1,4073	14	2	2	10,8
74,18	-1,13561	-30,1081	7	4	4	5,61
75,037	-6,94044	-36,6489	1	5	5	7,65
51,116	-9,64016	-56,2872	4	7	0	9,7
92,671	-0,3948	-53,3212	1	7	7	6,79
67,285	0,693898	-13,9139	6	3	3	4,92
95,559	1,087641	-37,2413	0	9	9	4,34
92,053	-50,471	-24,0934	5	5	3	5,2
72,016	-1,62379	-13,5036	0	7	1	4,85
66,993	-1,34345	-13,3839	0	7	1	3,37
95,258	-3,97625	-22,1769	10	2	1	7,52
62,058	-5,129	-45,3503	0	6	4	6,16
69,991	-40,4084	-40,8401	10	9	5	9
92,54	-1,25295	-34,7499	1	4	4	6,49
83,198	-9,59992	-18,843	8	3	3	2,2
97,323	0,106648	-35,8011	7	6	6	4,31
59,821	-29,5932	-25,0542	4	4	3	6,22
70,298	-16,781	-34,5749	8	4	3	5,19
75,451	-63,997	-19,9846	8	7	7	6,7
91,505	-12,0549	-19,4588	4	4	4	3,85
90,141	-1,78583	-16,4877	3	1	0	2,93
90,455	-2,26381	-21,1884	3	2	2	3,37
92,192	-0,57772	-19,3901	0	2	0	1,96
89,546	-0,95078	-18,5163	2	3	0	1,49
96,884	-10,3534	-20,8277	3	0	0	5,3
83,082	-21,9591	-34,6154	6	4	4	3,86
98,06	-27,9141	-22,9057	2	3	1	3,89
79,132	-2,45686	-21,7714	6	3	3	8,69
97,118	-8,0158	-40,628	2	14	14	5,43
92,939	-1,14863	-36,989	1	2	1	8,69

B	C	D	E	F	G	Saida
86,089	-20,4268	-29,4199	10	5	4	3,42
69,197	-57,7067	-29,2236	2	5	4	6,7
96,604	-16,159	-16,5319	9	4	4	5,82
24,144	-5,62697	-3,29276	7	3	3	4,82
88,3	-0,55565	-20,3327	5	1	0	4,74
95,753	-20,7043	-17,8391	12	5	4	1,54
86,431	-12,2854	-20,9233	7	4	4	3,8
58,582	-27,0775	-25,5731	5	4	4	2,37
23,863	-4,85332	-5,80508	6	5	5	3,21
95,754	-24,802	-19,0222	8	3	3	5,4

Dados de Consumo de Energia

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	2	0	1	6	4	1	1	1	2	7590686	672714	36,7
0	2	0	1	3	5	1	1	2	2	7590736	667024	39,3
0	2	0	1	5	6	1	1	2	6	7598496	663376	47
0	2	0	1	1	6	1	1	1	1	7594258	670794	49
0	2	0	2	2	3	1	1	1	1	7593257	670485	58,7
0	2	1	1	3	6	1	1	1	2	7595299	669527	59
0	2	1	1	2	5	1	1	1	3	7590993	666769	65,7
0	2	0	1	1	5	1	1	1	3	7590157	671221	68
0	2	0	2	3	4	1	1	2	1	7593099	670528	68,3
0	2	1	1	6	6	1	1	1	5	7588944	669452	70,3
0	2	0	1	4	5	1	1	2	2	7594430	667759	72
0	2	0	1	2	5	1	1	2	4	7591227	666661	75,3
0	1	1	1	4	4	1	1	2	3	7598472	661499	75,7
0	2	0	1	4	3	1	1	1	2	7598277	661681	77
0	2	0	1	4	6	1	1	1	2	7595716	673169	81
0	2	1	1	2	6	1	1	1	2	7594804	670180	86,7
0	2	0	1	1	4	1	1	1	2	7595242	669457	90
0	2	0	1	4	5	1	1	1	3	7598805	661137	97
0	2	0	2	2	7	1	1	1	4	7597390	667687	97
1	1	1	1	1	6	1	1	1	4	7594214	671122	97,3
0	2	1	1	4	3	1	1	1	4	7595071	672469	99,3
0	2	1	1	4	7	1	1	1	3	7590762	672666	101,3
0	2	0	1	4	4	1	1	1	3	7590870	669241	102,7

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	2	1	1	6	5	1	1	1	2	7597244	665926	102,7
0	2	0	1	2	5	1	1	1	3	7589109	669447	103
0	2	0	1	4	6	1	1	2	8	7600326	670629	110
0	2	0	3	2	6	1	1	1	6	7590884	667048	111
0	1	1	1	5	7	1	1	1	8	7594620	667117	113
0	1	0	1	3	5	1	1	1	3	7596199	671491	113,3
0	1	1	1	4	8	1	1	2	2	7590729	672641	124,3
0	2	0	1	1	5	1	1	1	6	7598564	661152	124,7
0	2	0	2	2	5	1	1	1	4	7597512	664662	126
0	2	0	1	1	4	1	1	1	5	7593078	667650	128
0	2	0	1	1	4	1	1	1	4	7592936	667720	128,3
0	2	0	1	6	4	1	1	1	3	7588746	670983	128,7
0	2	0	2	2	6	1	1	1	2	7594675	670279	133,3
0	2	0	1	1	4	1	1	1	5	7594273	670611	134,3
0	2	0	1	1	5	1	1	1	5	7592786	667759	138,3
0	2	1	1	2	6	1	1	1	4	7593201	671763	139,7
1	2	0	1	3	7	1	1	1	2	7594733	670594	141,3
0	2	2	1	4	6	1	1	1	4	7594227	670027	142,7
0	2	1	1	4	7	1	1	2	3	7590759	672819	144,7
0	2	1	1	2	4	1	1	1	5	7599034	661734	147
0	2	1	1	1	5	1	1	1	5	7594105	670690	148,7
0	1	1	1	1	5	1	1	1	2	7595265	668748	149,3
0	2	0	1	1	5	1	1	1	5	7598339	661410	150,7
0	2	0	1	3	8	1	1	2	3	7590880	669324	156
0	2	1	2	2	5	1	1	1	5	7597509	664653	156
0	2	1	1	3	7	1	1	2	3	7595441	668820	158

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	2	0	1	1	4	1	1	1	4	7595140	670332	162,7
0	2	0	1	2	5	1	1	1	4	7597077	665975	168
0	2	0	1	1	5	1	1	1	7	7590114	671085	170
0	2	0	3	3	6	1	1	1	6	7590665	669385	173
0	2	0	1	1	4	1	1	1	3	7596100	671654	174
0	2	0	2	2	5	1	1	2	6	7595218	668769	179
0	1	0	1	4	5	1	1	1	3	7600765	661789	185,7
0	2	0	2	2	5	1	1	1	5	7588993	670934	194,7
0	2	0	1	3	6	1	1	1	5	7588871	669409	196
0	2	1	1	3	5	1	1	2	4	7595304	668858	197,7
0	2	1	1	4	5	1	1	2	4	7600421	670966	221,3
0	2	0	1	2	8	1	1	1	4	7590128	671304	236,7
0	2	2	1	2	5	1	1	2	4	7588757	665283	262,7
0	2	0	1	2	4	1	1	3	3	7593500	669846	626,7
0	2	0	1	2	7	2	1	1	1	7594171	669810	60,7
0	2	0	1	4	3	2	1	1	3	7594784	664734	64
0	2	1	2	2	7	2	1	1	2	7595425	672572	73
0	2	0	2	2	7	2	1	1	5	7591326	667107	73
1	1	1	1	2	6	2	1	1	2	7595459	671292	73,3
0	2	0	3	2	4	2	1	1	1	7595332	668865	85
0	2	1	1	4	5	2	1	2	1	7590759	672819	95,7
0	2	1	1	1	3	2	1	1	2	7590594	672855	101
1	2	1	1	2	5	2	1	1	4	7598416	661447	125,7
0	2	0	1	4	7	2	1	1	1	7594115	669870	131,7
0	2	1	1	6	10	2	1	2	3	7590758	669321	140,3
0	2	0	3	2	7	2	1	2	5	7590720	669303	141,3

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	2	0	2	3	8	2	1	2	3	7594735	670364	144
0	2	0	2	1	7	2	1	2	3	7593117	671465	146,3
0	2	0	1	2	4	2	1	1	3	7592778	667815	146,7
0	2	0	1	3	7	2	1	1	5	7589027	669396	151,7
0	2	1	1	3	8	2	1	1	5	7590960	666672	153,7
0	2	0	1	3	7	2	1	2	5	7590943	666787	160,3
0	2	2	1	6	6	2	1	1	5	7590086	671184	165,7
0	1	1	1	6	12	2	1	2	5	7594321	666200	174,7
0	2	0	1	2	9	2	1	1	6	7593143	671700	176,7
0	2	0	1	3	4	2	1	1	5	7590706	672622	188
0	2	1	1	3	8	2	1	2	3	7590933	667017	193
0	1	5	1	2	8	2	1	1	4	7592032	666868	194,3
0	2	2	1	3	10	2	1	2	4	7594185	670758	199,3
0	2	0	1	4	7	2	1	2	8	7594428	667505	203,3
0	2	0	2	2	8	2	1	1	4	7597545	664586	205,7
0	2	0	1	4	5	2	1	1	3	7596125	671311	211,3
0	2	0	1	5	7	2	1	1	5	7591663	670063	228,7
0	2	0	1	6	10	2	1	2	2	7594857	666477	245
0	2	0	1	3	8	2	1	2	8	7598381	661465	246
0	2	1	1	4	5	2	1	2	4	7590728	672894	252,3
0	2	2	1	5	6	2	1	2	6	7594516	666876	256,7
0	2	0	1	4	9	2	1	1	5	7588793	665320	271,7
0	2	2	1	6	6	2	1	2	4	7590680	672885	298,7
0	2	1	1	2	8	2	1	2	13	7593148	671768	321,3
0	2	1	1	4	14	3	1	1	3	7594767	664735	186
0	2	1	1	4	8	3	1	1	2	7591664	670134	190

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
1	2	1	3	4	12	3	1	1	4	7590898	671530	205,3
0	1	1	1	3	6	3	1	2	1	7594208	670209	344,7
0	2	2	1	4	6	4	1	2	3	7594311	667039	150
0	2	0	1	1	7	1	0	1	7	7588843	670956	110,3
0	2	1	1	3	5	2	0	1	3	7594791	664730	104,3
0	2	0	1	3	8	2	0	1	6	7588941	670858	147,7
0	3	0	1	6	6	1	1	1	1	7590742	672813	37,3
0	3	1	1	2	5	1	1	1	2	7600425	670868	62,3
0	3	0	1	2	6	1	1	1	3	7595522	668683	71,7
0	3	1	1	4	7	1	1	1	2	7591412	669996	73,7
0	3	0	2	2	6	1	1	1	1	7597169	670160	74,3
0	3	0	1	3	5	1	1	1	2	7594494	667512	82,7
0	3	0	2	4	5	1	1	1	2	7588808	671014	83,3
0	3	0	1	1	4	1	1	1	3	7593198	667804	85,3
1	3	1	1	4	6	1	1	1	3	7597337	665965	92
0	3	1	1	4	5	1	1	2	2	7597312	665989	94,3
0	4	0	1	1	4	1	1	1	3	7592838	667785	98
0	3	1	1	2	6	1	1	1	7	7598967	661685	98,7
0	4	1	1	2	5	1	1	1	4	7590692	667028	99,7
0	3	0	1	2	5	1	1	1	3	7594783	670515	101,7
0	3	0	1	2	3	1	1	2	3	7598812	661870	102
0	3	1	1	4	4	1	1	1	2	7597987	661042	102,3
0	3	0	2	2	7	1	1	1	4	7594716	670326	112
0	4	1	1	2	4	1	1	2	2	7594734	664718	114
0	4	0	2	3	6	1	1	1	3	7593144	671688	117,3
0	3	0	1	4	4	1	1	1	2	7591669	670084	119

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	3	0	1	2	3	1	1	1	3	7594343	667652	122
0	3	0	1	1	4	1	1	1	3	7595040	670273	122,3
0	3	1	1	1	5	1	1	1	4	7592030	666716	123,3
0	3	1	1	4	4	1	1	1	4	7591238	667094	123,7
0	4	1	1	2	5	1	1	1	4	7598635	663425	124,3
0	4	0	2	1	6	1	1	1	2	7590015	671156	127,3
0	3	0	1	3	6	1	1	1	6	7593335	669618	130,7
0	4	0	1	4	7	1	1	1	3	7600877	661840	133,3
0	3	0	1	3	4	1	1	1	3	7596174	671620	143
0	4	0	1	6	5	1	1	1	4	7597279	665942	143,7
0	4	1	1	2	6	1	1	1	4	7593238	671406	161,3
0	4	0	1	3	6	1	1	1	4	7594488	667636	162,3
0	3	0	1	1	6	1	1	1	3	7599041	661728	170,3
0	3	1	1	3	6	1	1	1	3	7589383	670712	173
0	3	1	1	6	5	1	1	2	3	7594481	667432	180
0	3	0	2	3	5	1	1	2	3	7596954	670503	181
0	3	1	1	2	5	1	1	1	3	7595445	669695	188,7
0	3	0	1	4	6	1	1	1	3	7599061	661760	207
0	3	0	1	2	4	1	1	1	7	7598917	661824	210
0	3	1	1	2	5	1	1	2	5	7594625	667174	224,7
0	3	1	1	3	5	1	1	2	5	7590660	672878	225,3
0	3	0	1	2	5	1	1	1	2	7593412	669622	229,7
0	3	0	1	2	6	1	1	1	4	7589224	670762	236,7
0	3	0	2	1	6	1	1	1	3	7597469	664623	240,3
0	3	1	1	2	7	1	1	2	2	7601135	671066	350,7
0	3	0	1	6	6	2	1	2	3	7598528	663318	36

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	3	0	1	3	10	2	1	1	3	7600440	662154	54,3
0	3	0	1	3	9	2	1	2	3	7588855	669409	58,3
0	3	1	1	4	8	2	1	1	2	7594791	670529	77,3
0	3	0	1	4	7	2	1	1	3	7594771	664727	79,3
0	3	0	1	2	8	2	1	1	5	7595019	669673	93
0	3	1	2	2	8	2	1	2	2	7590183	671257	116,7
0	3	1	2	3	8	2	1	2	3	7592257	670703	128,3
0	3	0	1	3	7	2	1	2	3	7594922	669666	140
0	3	2	1	2	8	2	1	1	4	7594225	670752	143,3
0	4	0	1	5	7	2	1	1	5	7589007	669373	146,7
0	3	1	1	2	5	2	1	1	3	7590156	671117	148
0	3	1	1	3	5	2	1	1	6	7597101	670489	148,7
0	3	1	1	3	6	2	1	2	6	7594127	668950	158
0	4	1	1	3	8	2	1	2	4	7591321	666912	159,7
0	3	0	1	3	7	2	1	1	5	7594234	670310	162,7
0	4	2	1	3	9	2	1	2	5	7590062	671214	163
0	3	0	1	6	8	2	1	1	6	7598721	663347	164,7
0	3	1	1	6	8	2	1	2	4	7598788	663483	430,3
0	3	0	1	6	8	3	1	2	2	7597110	665979	130,3
0	3	1	1	4	7	3	1	2	2	7591059	666585	139,7
0	4	1	1	4	5	3	1	2	3	7595480	669484	180
0	3	1	1	4	9	3	1	2	5	7590784	672510	190
0	4	2	1	3	10	3	1	1	7	7592115	667005	219
0	3	1	1	6	12	3	1	2	4	7589312	670775	308,7
0	4	0	1	1	6	4	1	1	4	7590831	669245	145
1	3	1	1	6	9	6	1	3	6	7589399	666362	379

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	4	1	1	2	6	1	0	1	5	7594811	664769	107
0	3	0	1	2	5	1	0	1	2	7590619	669178	123,3
0	3	1	1	6	6	1	0	1	5	7588929	669362	153,7
0	3	1	2	3	13	1	0	2	4	7595400	668842	266
0	3	1	1	5	6	1	0	2	4	7598814	661992	366,7
0	3	0	1	2	4	3	0	1	5	7594799	664730	28
0	5	0	1	2	3	1	1	1	2	7598960	661684	32,3
0	5	0	1	4	8	1	1	1	3	7601057	661866	50,7
0	5	0	1	3	7	1	1	1	2	7591483	669893	58
0	5	0	1	4	5	1	1	1	2	7595187	672492	70,3
0	5	1	2	2	6	1	1	1	2	7597116	670051	71,3
0	5	1	1	5	8	1	1	1	4	7597101	670490	74,7
0	5	1	1	2	6	1	1	1	2	7595421	671222	81
0	5	0	1	4	8	1	1	1	3	7594851	670621	82
0	5	2	1	3	7	1	1	2	3	7591287	666801	82,7
0	5	0	2	2	7	1	1	1	2	7591102	671061	86,7
0	5	0	1	6	8	1	1	1	2	7596036	671417	88,7
0	5	0	2	1	5	1	1	2	3	7595489	669625	90,3
0	5	0	3	3	6	1	1	1	2	7593209	671777	92,7
0	5	1	2	3	6	1	1	2	4	7590660	669250	95,3
0	5	2	1	2	4	1	1	1	4	7596281	671623	102
0	5	1	2	3	5	1	1	1	2	7593246	670462	107
0	5	0	1	2	6	1	1	1	3	7594610	666293	107,7
0	5	0	1	2	7	1	1	1	4	7591929	666564	112,3
0	5	0	2	1	6	1	1	1	4	7594117	670663	113,3
0	5	0	1	1	5	1	1	1	5	7594317	670718	118,3

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	5	0	1	1	4	1	1	1	3	7594651	669408	127
0	5	0	1	4	7	1	1	1	4	7600421	670966	129,7
0	5	0	1	3	5	1	1	1	5	7594533	667528	132
0	5	0	1	1	5	1	1	1	3	7594626	669764	134
0	5	0	1	2	8	1	1	1	4	7600602	661776	148
0	5	0	1	4	6	1	1	1	5	7590614	669358	158,7
0	5	0	2	2	5	1	1	1	5	7593354	667818	162,3
0	5	2	2	3	5	1	1	2	3	7594970	670239	162,3
0	5	0	1	3	6	1	1	1	1	7593406	669490	163
0	5	0	2	2	4	1	1	1	3	7597468	664525	163
0	5	1	1	4	10	1	1	2	5	7590771	669250	168,3
0	5	1	2	2	6	1	1	1	3	7594932	669684	170,7
0	5	0	2	1	6	1	1	1	7	7593100	671660	177,7
0	5	2	1	5	7	1	1	2	5	7598139	663287	184,3
1	5	0	1	3	9	1	1	2	2	7593269	671335	201,3
1	5	1	1	2	6	1	1	1	6	7594155	671152	221
0	5	1	2	4	8	1	1	2	4	7594234	670310	223,7
0	5	1	1	6	6	1	1	2	3	7588900	665188	249,7
1	5	1	1	2	6	1	1	1	6	7595348	668738	260
0	5	1	2	2	5	1	1	1	4	7591860	669944	280
0	5	0	1	2	5	1	1	2	7	7594386	667119	283,7
0	5	1	1	4	8	1	1	3	4	7600964	661720	301,3
0	5	3	2	2	5	1	1	1	4	7591696	669821	363
0	5	0	1	6	5	2	1	1	2	7591531	666900	53
0	5	1	1	6	6	2	1	1	2	7594603	667028	66
0	5	1	1	6	12	2	1	2	2	7594569	666833	66,7

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	5	0	1	2	7	2	1	1	3	7594405	666965	69,7
1	5	1	1	3	6	2	1	2	1	7597026	670351	73
0	5	1	1	5	9	2	1	2	2	7594775	666558	83
0	5	0	1	3	8	2	1	1	2	7591664	666841	88,7
0	5	1	2	6	7	2	1	1	2	7590540	669303	88,7
0	5	1	1	5	8	2	1	1	4	7594986	666478	92,3
0	5	0	1	6	9	2	1	2	2	7591626	670184	94
0	5	0	1	2	8	2	1	2	4	7592368	667240	94,7
0	5	0	1	6	7	2	1	1	3	7591704	670005	100,3
1	5	0	1	2	6	2	1	1	4	7595595	673269	103
0	5	1	1	3	5	2	1	1	3	7594091	670187	103,7
0	5	1	1	5	9	2	1	2	3	7595436	668758	120,7
0	5	0	1	5	9	2	1	1	5	7594330	670768	127,3
0	5	0	2	2	7	2	1	1	5	7597420	664733	128,3
0	5	0	1	4	7	2	1	2	4	7594909	669760	135,7
0	5	1	1	4	8	2	1	2	3	7594533	667212	141,7
0	5	0	1	5	7	2	1	1	4	7593407	669686	151
0	5	2	1	6	10	2	1	1	4	7594642	666778	151,3
0	5	1	2	2	9	2	1	1	2	7590948	671566	155,3
0	5	1	2	5	7	2	1	2	5	7593684	669439	156
0	5	0	1	5	8	2	1	1	4	7594494	667360	159,3
0	5	0	1	3	7	2	1	2	5	7589066	670848	167,3
0	5	0	2	2	5	2	1	2	3	7593099	670528	173,7
0	5	1	1	3	8	2	1	2	6	7597028	665987	184,7
0	5	1	2	3	6	2	1	2	4	7594078	668986	189
0	5	1	1	6	11	2	1	2	4	7594399	666170	198

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
1	5	0	2	4	12	2	1	1	2	7592197	670754	202,7
0	5	1	1	4	6	2	1	1	5	7590744	672657	203
0	5	1	2	3	8	2	1	2	6	7596973	670439	204,7
0	5	1	1	5	6	2	1	2	2	7599376	662099	206,7
0	5	2	1	3	9	2	1	2	5	7594300	670288	211,7
0	5	1	1	3	8	2	1	1	5	7601058	661932	213,3
0	5	0	1	2	8	2	1	1	3	7595372	668788	216,3
0	5	1	2	2	6	2	1	2	4	7591791	669940	219
1	5	1	1	2	6	2	1	2	4	7594313	667654	245
0	5	0	2	2	5	2	1	1	4	7594688	670437	246,7
0	5	1	1	3	6	2	1	3	5	7595526	669620	260
0	5	1	1	2	6	2	1	1	5	7593964	670791	269
0	5	0	2	6	8	2	1	1	7	7593014	669935	275,3
0	5	1	2	5	5	2	1	3	5	7593598	669337	301,7
1	5	2	2	1	5	2	1	2	1	7593613	669500	319,7
0	5	1	3	4	8	2	1	2	4	7591671	669787	390
0	5	1	1	3	7	3	1	1	3	7598706	663433	95,3
0	5	1	2	3	7	3	1	2	4	7593628	669335	132,7
1	5	0	1	4	10	3	1	1	2	7594778	670423	165
0	5	2	2	6	8	3	1	2	4	7597119	670129	166
0	5	1	2	4	6	3	1	2	4	7594170	670090	199
0	5	0	1	3	9	3	1	2	5	7590909	671520	257,7
1	5	2	1	6	9	3	1	2	5	7594871	670261	300
0	5	0	2	6	10	4	1	3	3	7591541	670921	225,7
1	5	3	1	6	20	4	1	3	3	7590965	671379	319,3
1	5	1	1	5	9	4	1	3	4	7588406	666047	541,7

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
1	5	0	3	6	7	4	1	3	17	7593525	669994	665,7
0	5	1	1	1	4	1	0	2	4	7594002	668842	76,7
0	5	1	2	2	7	1	0	1	1	7593063	669966	93
0	5	1	2	2	10	2	0	2	4	7593094	670041	152,3
0	7	0	2	3	4	1	1	1	1	7594069	670042	49,3
0	7	0	1	3	9	1	1	1	1	7591411	669994	69,3
0	7	0	2	2	2	1	1	1	1	7593257	670485	78,7
0	7	0	2	1	3	1	1	1	2	7593100	670494	88
0	7	1	2	2	5	1	1	2	2	7594155	670018	93,7
0	7	0	2	2	6	1	1	1	5	7598827	663422	130
0	7	2	2	2	8	1	1	1	4	7597441	664678	146,7
0	7	1	1	3	6	1	1	2	3	7591383	669764	162
0	7	1	2	5	6	1	1	2	3	7593167	669835	169,3
0	7	0	1	2	5	1	1	1	1	7594084	670008	186,3
0	7	0	1	4	9	1	1	2	4	7598261	661644	217,7
1	7	1	1	3	5	2	1	1	3	7593152	671714	67
1	7	0	1	1	10	2	1	1	1	7593188	671492	80,7
0	7	0	1	4	8	2	1	2	3	7594801	670575	84,7
0	7	0	1	3	7	2	1	1	2	7591427	669846	114,3
0	7	1	2	3	6	2	1	2	2	7591785	669932	122
0	7	0	2	3	6	2	1	1	2	7594246	669898	132
0	7	0	1	4	5	2	1	1	3	7594768	670643	140,7
1	7	3	1	3	6	2	1	3	3	7592174	670706	141,3
1	7	1	2	4	11	2	1	2	2	7593074	670596	142,7
0	7	0	2	2	8	2	1	1	4	7593226	670010	149
0	7	1	1	3	6	2	1	2	2	7597104	670108	152

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	7	2	2	3	6	2	1	2	3	7593134	671549	152,3
0	7	1	1	4	10	2	1	1	2	7591480	670172	155,3
0	7	1	1	4	10	2	1	1	2	7591398	669940	157,3
1	7	0	1	4	6	2	1	1	3	7594549	666935	162
0	7	1	1	4	7	2	1	2	3	7591425	670037	165
0	7	1	1	4	7	2	1	1	5	7595384	669494	172
1	7	1	1	2	8	2	1	1	2	7591078	671185	176,7
0	7	1	1	2	12	2	1	2	3	7594168	671031	181
0	7	1	1	4	7	2	1	2	1	7593060	671460	188,7
0	7	2	2	5	7	2	1	2	5	7591656	670063	192
0	7	1	1	3	7	2	1	2	3	7594817	670275	196
0	7	1	2	3	6	2	1	2	3	7593527	669468	196,7
1	7	0	2	1	4	2	1	2	3	7594953	670283	198,3
0	7	1	2	3	6	2	1	1	5	7594179	669882	227
1	7	3	1	4	7	2	1	2	6	7594968	670270	232,3
1	7	2	2	2	10	2	1	1	3	7593118	669941	237,3
0	7	0	1	3	5	2	1	2	4	7593863	671721	249,7
1	7	1	2	3	7	2	1	1	4	7591697	669822	251
0	7	3	1	5	7	2	1	2	4	7594867	670277	263,3
0	7	1	1	6	5	2	1	2	4	7588781	666176	266,3
0	7	0	2	2	9	2	1	2	3	7591136	671187	274,3
0	7	1	1	2	8	2	1	1	2	7594902	670328	279
0	7	1	1	4	10	2	1	2	3	7593070	667736	281
1	7	1	1	3	16	2	1	2	6	7591414	669812	360
0	7	0	2	2	7	2	1	2	5	7593099	670528	374
1	7	1	2	3	12	2	1	2	4	7591130	671114	384

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
1	7	2	2	5	13	3	1	1	4	7593390	669702	56
0	7	0	2	2	7	3	1	2	2	7591813	669966	143,7
0	7	2	1	3	6	3	1	2	3	7594083	668636	189,3
1	7	1	1	4	10	3	1	1	5	7598472	663284	205,7
0	7	0	1	4	10	3	1	3	1	7594147	671118	227,3
1	7	3	2	5	7	3	1	2	4	7592158	670779	233,3
0	7	2	1	6	9	3	1	2	4	7591698	669983	292,3
1	7	2	1	6	16	3	1	3	3	7593196	669909	294,3
1	7	1	1	3	7	3	1	3	6	7593248	670468	301
0	7	2	1	5	10	3	1	3	4	7591098	671207	325
1	7	2	1	6	16	3	1	2	4	7593236	669839	342,7
1	7	2	1	6	9	3	1	3	3	7594824	670195	388,7
1	7	2	2	4	7	3	1	2	3	7593163	670421	435
1	7	3	1	6	18	3	1	3	4	7592174	670644	437,7
1	7	1	1	5	8	3	1	3	6	7588590	666014	466
1	7	4	1	6	10	4	1	1	4	7593376	669664	67
1	7	1	2	3	20	4	1	3	3	7592273	670735	184,7
1	7	3	1	5	11	4	1	2	5	7588715	666149	473,7
1	7	2	1	6	16	4	1	3	4	7592257	670661	535,3
0	7	1	1	6	11	5	1	2	4	7593865	668756	145,7
1	7	3	1	5	13	5	1	3	5	7588483	666044	384,7
1	7	1	2	6	16	6	1	3	3	7592262	670758	330,3
0	7	3	2	6	15	6	1	2	5	7590528	669340	364,3
1	7	3	1	6	11	7	1	2	5	7591696	670115	326,7
0	7	1	1	1	11	2	0	1	2	7592215	666649	127,7
0	6	0	1	1	5	1	1	1	3	7594241	670889	78,3

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	6	0	2	2	3	1	1	1	2	7593152	670414	83,3
0	6	0	2	1	6	1	1	1	2	7598726	663711	99,3
0	6	0	1	5	8	1	1	1	4	7594532	667343	142
0	6	0	1	2	7	1	1	1	4	7594235	670928	186
0	6	0	1	5	7	1	1	1	5	7594617	666414	270,7
0	6	2	1	5	7	1	1	1	5	7594575	666275	382,7
0	6	0	2	2	9	2	1	2	2	7592090	670703	92
0	6	2	1	6	11	2	1	1	3	7594602	666926	112,7
1	6	1	1	3	9	2	1	1	2	7594731	670349	129
0	6	0	1	5	5	2	1	3	2	7589246	666167	171,7
0	6	0	2	1	6	2	1	1	4	7594732	670259	183
0	6	0	1	1	7	2	1	2	5	7594236	671054	186,7
1	6	1	1	4	9	2	1	1	4	7593237	669955	211,3
1	6	0	2	1	5	2	1	1	5	7592108	670732	232,3
0	6	1	1	6	8	3	1	2	3	7594047	668804	181,7
1	6	3	1	6	14	3	1	2	4	7594305	666308	349,3
0	6	1	1	2	12	3	1	1	5	7591024	671218	361,3
0	6	0	1	3	5	1	0	2	3	7591550	666984	113
0	6	0	1	5	8	2	0	2	5	7594640	667199	311,7
0	2	0	1	5	6	1	1	1	4	7600288	671001	145,7
0	2	0	1	2	6	1	1	1	5	7598632	669120	169,7
1	2	1	2	3	8	2	1	2	3	7593265	670502	299
0	3	0	1	2	6	1	1	1	2	7590594	669101	86,7
0	3	0	1	2	6	1	1	1	3	7594222	670238	139,3
0	3	1	1	6	8	1	1	1	3	7594548	666333	141
0	3	0	1	2	6	1	1	1	9	7595547	671260	203,3

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	3	2	1	4	7	1	1	1	3	7590695	672691	224,3
0	3	1	2	4	6	1	1	1	3	7590645	672835	226,7
0	3	1	1	3	5	2	1	2	2	7597091	670475	172
0	5	0	1	2	4	1	1	1	3	7597236	670291	54,3
0	5	1	2	2	5	1	1	1	1	7597478	664711	132
0	5	0	1	4	7	1	1	2	2	7595794	673116	136,3
0	5	0	1	4	6	2	1	1	4	7601199	671145	124,3
0	5	1	1	5	8	2	1	1	3	7594991	666503	154
0	5	2	1	4	8	2	1	3	2	7593430	669886	276,3
0	5	1	1	6	10	3	1	2	3	7591379	669754	153,7
0	5	1	1	3	5	3	1	1	5	7594193	670027	287,7
0	5	1	1	5	6	1	0	2	4	7592968	667811	319,7
0	7	0	2	2	6	1	1	1	2	7593184	671478	74,7
0	7	1	2	2	9	2	1	1	3	7591482	670082	226,3
0	7	1	1	4	2	3	1	2	4	7590887	667119	278
0	7	2	1	6	9	3	1	3	7	7594115	670007	482,3
0	6	1	2	2	9	2	1	1	4	7593162	671430	162
0	2	0	1	1	4	1	1	1	2	7598339	661410	63,3
0	2	0	3	2	6	1	1	1	4	7590628	669231	105,3
0	2	0	1	2	4	1	1	2	5	7594540	667021	117
0	2	0	1	2	6	1	1	1	3	7591276	666808	122,3
0	2	0	1	2	5	1	1	1	3	7594480	667288	136,7
1	1	1	1	1	5	1	1	1	3	7594295	670689	157,3
0	2	0	1	3	8	1	1	1	16	7592044	666518	163
0	2	0	2	2	7	1	1	1	9	7597438	664580	421,7
0	1	1	1	5	9	2	1	2	3	7594654	666831	188,3

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
1	2	1	1	6	6	2	1	1	3	7593146	667781	264,7
0	4	0	1	3	8	1	1	2	2	7594708	666558	108
0	3	0	2	1	4	1	1	1	2	7594246	670598	123,3
0	3	0	1	1	4	1	1	1	5	7592832	667740	126,3
0	4	2	1	4	6	1	1	1	6	7594793	664735	129,3
0	3	0	1	1	4	1	1	1	4	7595024	669731	154,3
0	4	1	1	5	9	1	1	1	3	7594630	666369	168,3
0	3	0	1	2	6	1	1	1	4	7600477	661721	178
0	3	0	1	5	6	2	1	1	3	7594763	664763	63
0	3	1	1	4	6	2	1	2	3	7594033	668949	140
0	5	0	1	5	7	1	1	1	4	7593372	669632	80,3
0	5	0	1	2	6	1	1	1	2	7594152	671178	118
0	5	0	1	2	6	1	1	1	3	7600478	662051	135
0	5	0	1	1	5	1	1	1	4	7599106	661802	158,3
0	5	0	2	2	4	1	1	2	3	7596952	670506	180,3
0	5	0	1	1	4	1	1	1	4	7593951	669011	191,3
0	5	0	1	3	8	2	1	1	1	7594988	669612	58,7
0	5	1	2	3	5	2	1	1	2	7594161	669878	81,3
0	5	1	1	4	7	2	1	2	3	7593950	668857	130,7
1	5	1	1	4	5	2	1	3	4	7591053	666706	268,7
0	7	0	2	2	6	1	1	1	2	7595387	669497	72,7
0	7	1	2	6	9	2	1	1	3	7592161	670787	156,3
0	7	3	1	5	7	2	1	1	3	7594713	670342	188
0	7	1	1	6	7	4	1	3	5	7593982	668564	292
1	7	1	2	5	11	4	1	3	4	7591173	671189	328
0	6	0	1	5	6	1	1	1	2	7595287	669389	67

empregd	escolaridade	carros	tipo	area	comodos	banheiro	relogio	c_fases	resident	coord_n	coord_e	Saida
0	6	1	1	5	5	2	1	1	2	7594633	667165	89,3
0	6	0	1	4	6	2	1	2	2	7594478	667331	90
0	6	0	2	1	6	2	1	1	4	7593159	669420	119,7
0	6	2	1	5	8	2	1	1	4	7594646	666427	185,3
1	6	1	1	6	8	7	1	3	4	7593683	669481	403,3

Dados de Imóveis - Apartamentos

pos. do apto	elev	gar	local	área	pav	andar	peças	salas	dorm	suíte	banh	dep. de emp.	dist. escola	dist. hosp.	dist. merc.	acab.	revest. préd	cons	idade real	idade aparen	Saida
3	2	1	1	222	15	3	9	1	2	1	1	1	2	2	3	3	4	2	2	2	130000
3	1	1	1	162,44	7	1	9	1	2	1	1	1	3	1	3	3	1	5	4	6	85000
3	1	1	1	176	8	2	9	1	2	1	1	1	3	2	3	3	1	5	4	6	80000
3	2	2	1	179,2	14	3	10	2	2	1	1	1	3	2	3	3	4	5	4	6	115000
3	2	2	1	279,4	20	3	8	1	2	1	1	1	3	2	3	3	4	5	2	4	150000
3	1	1	1	120	12	3	12	2	3	1	2	1	3	2	3	3	2	5	3	4	110000
2	1	1	1	220	16	2	12	3	3	1	2	1	2	2	2	3	4	4	2	2	120000
3	1	1	1	107	8	1	9	3	2	1	1	0	2	2	2	2	1	3	3	5	68000
2	0	1	1	50	4	2	4	1	1	0	1	0	3	2	3	2	1	4	4	4	40000
3	2	3	1	240	15	3	12	3	2	1	1	1	3	3	3	3	4	4	2	3	170000
3	0	1	0	100	3	2	7	2	3	0	2	0	3	1	1	2	1	3	2	3	50000
3	0	1	1	147	6	1	7	1	2	1	1	0	3	2	3	2	2	5	4	5	60000
3	1	1	1	108	7	3	11	2	2	1	1	1	3	3	3	2	4	3	3	3	93000
2	2	2	1	311	16	3	10	3	2	1	1	1	1	1	1	3	4	5	4	4	250000
2	1	1	1	132	7	2	8	2	2	1	1	0	3	2	1	2	1	4	5	5	65000
3	2	1	1	107	8	3	8	2	2	1	1	0	3	3	3	2	4	4	3	3	65000
3	2	1	1	107	8	4	8	2	2	1	1	0	3	3	3	2	4	4	3	3	71000
3	2	4	1	330	15	3	17	3	3	1	3	1	2	1	2	2	4	4	3	2	220000
3	2	2	1	220	15	3	13	2	3	1	1	1	3	3	3	2	4	4	5	5	200000
3	1	1	1	130	6	3	11	2	3	1	1	1	3	3	3	2	3,5	4	3	4	90000
3	2	2	1	164	10	3	11	2	3	1	1	1	3	3	3	2	3,5	4	3	4	140000
3	2	2	1	160	15	3	11	2	3	1	2	1	3	3	3	2	4	4	3	4	250000
3	2	1	1	374	15	3	14	3	2	1	3	1	1	1	1	2	4	4	4	4	250000

pos. do apto	elev	gar	local	área	pav	andar	peças	salas	dorm	suíte	banh	dep. de emp.	dist. escola	dist. hosp.	dist. merc.	acab.	revest. préd	cons	idade real	idade aparen	Saida
3	1	1	1	220	16	4	14	3	3	1	2	1	3	3	3	3	4	4	3	4	150000
2	1	1	1	220	16	3	14	3	3	1	2	1	3	3	3	3	4	4	3	4	120000
3	1	2	1	180	13	3	13	2	3	1	1	1	3	1	3	3	4	3	4	4	180000
3	1	2	1	320	13	3	11	1	3	1	1	1	3	3	3	3	4	4	5	5	250000
3	1	2	1	180	13	3	13	2	3	1	1	1	3	1	3	3	4	3	4	4	180000
3	1	2	1	320	13	3	11	1	3	1	1	1	3	3	3	3	4	4	5	5	250000
1	2	2	1	260	14	3	11	2	2	1	1	1	3	3	3	2	4	3	2	2	115000
2	2	2	1	260	14	2	11	2	2	1	1	1	3	3	3	2	4	3	2	2	120000
3	2	2	1	260	14	1	11	2	2	1	1	1	3	3	3	2	4	3	2	2	140000
3	1	1	1	140	8	2	6	1	2	1	1	0	2	2	3	2	4	3	5	5	90000
1	1	1	1	130	7	3	7	2	2	1	1	0	3	3	3	2	1	3	4	4	65000
1	2	2	1	260	14	1	11	2	2	1	1	1	3	3	3	2	4	3	2	2	120000
1	2	2	1	310	16	2	11	2	2	1	1	1	1	1	1	3	4	4	4	4	210000
3	0	1	1	100	4	3	7	1	3	0	2	0	3	3	3	2	1	3	1	1	100000
3	0	1	1	70	3	2	5	1	2	0	1	0	3	3	3	2	1	3	1	1	70000
3	0	1	1	90	3	2	6	1	2	0	2	0	3	3	3	2	1	2	1	1	95000
3	1	1	1	38	8	2	4	1	1	0	1	0	3	3	3	2	1	3	3	4	30000
3	1	1	1	40	8	3	5	1	2	0	1	0	3	3	3	2	1	3	3	4	40000
3	1	1	1	170	7	2	9	2	2	1	1	0	3	3	3	2	4	4	4	6	100000
1	1	1	1	170	7	1	9	2	2	1	1	0	3	3	3	2	4	4	4	6	90000
3	1	1	1	170	7	3	9	2	2	1	1	0	3	3	3	2	4	4	4	6	110000

Dados de Imóveis - Casas

bairro	gar.	suíte	banh.	edícula	dist. mercado	área const.	área terreno	acab.	cob.	estr.	conserv.	piscina	dorm.	dep. emp.	lav.	peças	idade aparen.	Saida
5	1	1	2	1	3	183	500	1	4	3	1	0	2	1	1	8	2	120000
4	1	1	3	0	2	216	977	1	4	3	1	0	3	1	1	13	2	160000
4	1	1	3	0	2	155	420	2	4	3	1	0	3	1	1	12	2	110000
2	1	1	2	0	1	170	490	2	4	3	1	0	2	0	1	7	3	70000
4	1	1	2	1	1	160	480	2	4	3	3	0	2	0	1	9	3	85000
5	1	1	2	1	3	160	500	2	4	3	3	0	3	0	1	8	3	100000
5	1	1	2	1	3	134	1000	2	4	3	2	0	4	0	1	10	4	160000
3	1	0	1	0	3	70	300	1	2	1	1	0	3	0	0	5	1	30000
3	0	0	1	0	2	198	350	1	1	1	3	0	2	0	0	5	3	28000
3	1	0	1	0	2	113	400	2	2	1	2	0	3	0	0	8	2	35000
5	1	0	2	1	3	238	1200	2	4	3	2	0	4	0	1	9	1	150000
3	1	0	2	0	2	158	315	2	1	1	2	0	4	0	1	12	1	45000
3	1	0	1	0	3	124	300	2	4	3	2	0	3	0	0	8	1	30000
5	0	0	1	0	3	95	340	2	2	1	1	0	2	0	0	6	1	28000
4	1	1	1	1	3	187	490	3	4	3	2	0	2	1	1	12	4	140000
4	1	1	1	1	1	242	490	3	4	3	2	0	3	1	1	13	3	130000
3	0	0	2	0	1	100	480	2	4	3	2	0	3	0	1	8	5	50000
3	1	1	1	0	1	180	786	3	4	3	3	0	2	1	1	10	3	150000
5	1	1	3	1	1	380	480	2	3	3	2	0	3	0	1	14	5	150000
4	1	1	1	1	1	240	490	3	4	3	2	1	2	1	1	11	2	135000
3	1	1	1	1	2	184	1000	2	4	3	3	1	2	1	1	13	3	180000
3	1	0	2	0	3	140	446	2	1	2	2	0	3	0	1	6	2	40000
5	1	1	1	0	3	400	600	2	4	3	2	0	5	1	1	14	3	150000

bairro	gar.	suíte	banh.	edícula	dist. mercado	área const.	área terreno	acab.	cob.	estr.	conserv.	piscina	dorm.	dep. emp.	lav.	peças	idade aparen.	Saida
4	1	0	1	0	3	110	270	2	4	3	3	0	2	0	1	5	4	60000
2	0	0	1	0	3	120	300	2	4	3	1	0	3	0	1	6	3	15000
2	1	0	1	0	3	150	300	2	4	3	3	0	3	0	1	6	5	45000
5	1	1	2	0	3	400	750	2	4	3	3	0	3	0	1	11	3	165000
5	1	0	1	0	3	130	300	2	2	1	1	0	3	0	0	5	3	95000
4	1	1	1	0	3	200	400	2	4	3	3	0	3	0	0	9	6	100000
4	1	1	2	1	1	244	500	2	4	3	2	0	3	1	1	13	6	130000
5	1	1	1	1	1	113	300	2	4	3	2	0	2	1	0	9	4	90000
5	1	1	2	1	1	220	500	4	4	3	2	1	3	1	1	15	3	185000
5	1	1	1	1	3	92	650	2	4	3	3	0	2	0	1	7	5	90000
5	1	1	1	0	3	123	300	2	4	3	2	0	2	0	1	7	5	15000
5	1	0	1	0	3	150	1000	1	2	1	1	0	2	0	1	7	1	150000
5	1	1	2	1	3	200	400	2	4	4	2	0	2	1	1	7	2	150000
5	1	0	1	0	3	100	1000	1	1	1	1	0	2	0	1	5	1	140000
4	1	0	1	0	2	70	490	1	1	1	1	0	2	0	1	5	1	30000
5	1	0	1	0	3	100	950	2	2	1	1	0	3	0	1	7	1	60000
5	1	0	1	0	3	80	475	2	2	1	1	0	3	0	1	7	1	45000
3	0	0	1	0	2	70	600	1	3	3	3	0	2	0	0	5	5	30000
3	1	1	1	0	2	180	640	2	4	3	3	0	2	1	1	10	6	90000
2	0	1	2	0	2	70	480	2	4	3	3	0	1	0	1	8	5	50000
3	1	1	1	0	1	115	250	2	4	3	3	0	2	0	1	8	4	70000
3	1	1	1	0	1	160	450	2	4	3	3	1	2	1	1	9	1	130000
3	1	1	2	1	3	325	225	3	4	4	3	0	4	1	1	16	6	180000
5	1	1	2	0	3	320	450	3	4	4	3	0	2	0	0	13	4	290000
2	1	0	1	0	2	116	300	2	3	3	2	0	3	0	0	6	3	50000

bairro	gar.	suíte	banh.	edícula	dist. mercado	área const.	área terreno	acab.	cob.	estr.	conserv.	piscina	dorm.	dep. emp.	lav.	peças	idade aparen.	Saida
5	1	1	3	0	3	180	500	2	4	3	3	0	3	0,5	1	12	5	98000
2	1	0	1	0	3	100	800	2	3,5	3	2	0	2	0	0	5	4	65000
3	1	0	1	0	2	80	390	2	2	3	1	0	3	0	1	6	2	40000

Dados de Imóveis - Terrenos

localização	setor comer.	pólo	frente	área do ter.	proteção	plano	inclinado	posição	pavimentação	Saida
6	2	1	2	650	3	0	1	1	1	100000
2	0	0	3	640	3	1	0	2	1	25000
5	0	1	1	500	3	1	0	1	1	43000
3	0	1	2	390	3	2	0	1	1	33000
5	0	1	3	262	3	2	0	2	1	40000
5	2	1	3	1000	3	2	0	1	1	140000
4	1	1	3	950	3	2	0	1	1	90000
4	0	1	1	475	3	0	1	1	1	38000
5	1	1	3	470	3	3	0	2	1	70000
6	3	1	1	500	0	3	0	1	1	145000
2	0	0	2	420	0	3	0	2	0	15000
2	0	0	2	420	0	3	0	1	0	13000
3	2	1	3	2000	0	2	0	2	1	100000
4	1	1	3	950	3	2	0	1	1	90000
6	3	1	3	1000	3	2	0	2	1	250000
5	0	1	1	242	0	2	0	1	1	38000
6	3	1	3	940	2	3	0	2	1	300000
5	0	1	1	500	2	1	0	1	1	47000
5	0	1	3	350	0	2	0	2	1	60000
5	0	1	1	500	0	0	1	1	1	50000
3	0	0	2	336	0	3	0	1	1	15000
3	0	1	2	336	0	3	0	1	1	30000
4	0	1	2	300	0	2	0	1	1	25000
3	0	1	1	450	3	3	0	1	1	30000

Dados ABALONE

Esse banco de dados pode ser encontrado no seguinte repositório:

ftp://ftp.ics.uci.edu/pub/machine-learning-databases/abalone/